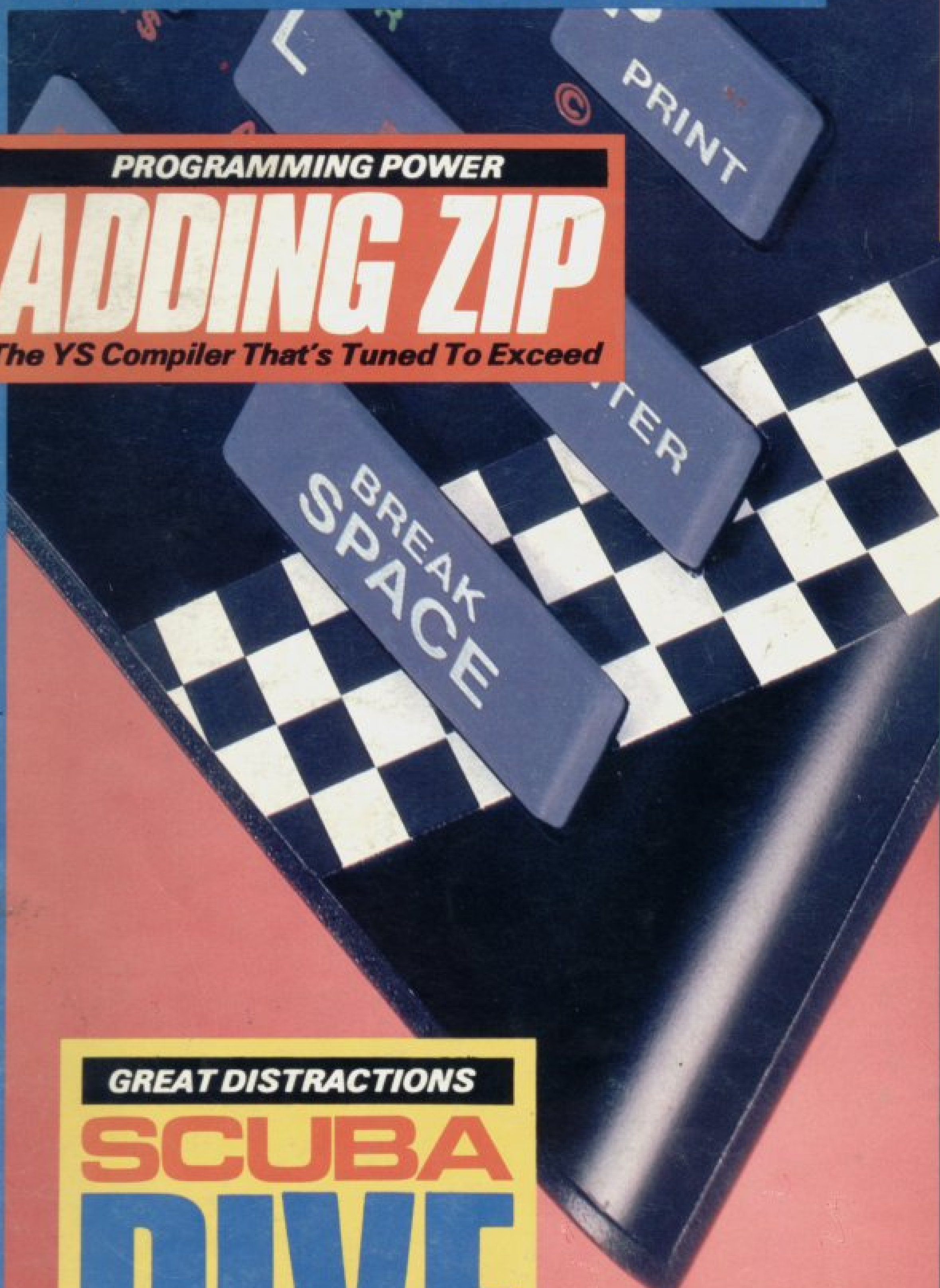
BRITAIN'S BRIGHTEST MICRO MAG NOW MONTHLY!

YOUR SPECTRUM & QL USER

85p No.3



INSIDE QL USER

SUPER BASIC

A Structural Survey

PROGRAMMING POWER

ADDING ZIP

The YS Compiler That's Tuned To Exceed

COMPETITION



WIN A STACK RIFLE

Twenty-Five Great Prizes!

GREAT DISTRACTIONS

SCUBA DIVE

Oceans Of Adventure

INSIDE YS

FEATURES



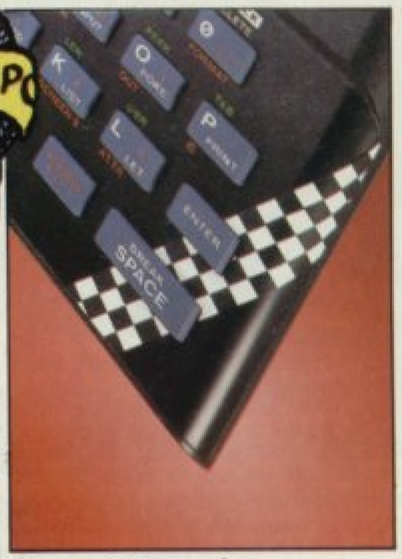


ON THE BENCH

QL USER



REGULARS



Cover photography by Ian McKinnell

Editor Roger Munford; **Managing Editor** Bruce Sawford; **Deputy Editor** Tina Boylan; **Editorial Assistant** Peter Shaw; **Software Consultant** Gavin Monk; **Sub Editor** Nik Lumsden; **Contributors** Phil Manchester, Ian Beardsmore, Ron Smith, Mike Mepham, Sandy Dewhurst, Colin Young, Andrew Wright, Richard Archdeacon, Stephen Adams, Damir Skrgatic, Dilwyn Jones, Simon Goodwin, Toni Baker, SQ Factor; **Art Editor** Jimmy Egerton; **Art Assistant** Steve Broadhurst; **Group Advertisement Manger** Jeff Raggett; **Advertisement Managers** Shane Campbell, Gill Harris, Jason Wood; **Production Editor** Derek Cohen; **Typesetters** Beverley Douglas, Maggie Kayley, Velma Miller; **Production Manager** Sonia Hunt; **Group Art Director** Perry Neville; **Publisher** Steve England; **Distribution Manager** Colin James; **Published by** Sportscene Specialist Press Ltd, 14 Rathbone Place, London W1P 1DE. Company registered in England. **Telephone (all departments)** 01-631 1433. **Telex** 8954139 BunchG. **Reproduction** Graphic Ideas, London; **Printers** Chase Webb Offset, St Austell, Cornwall; **Distribution** Seymour Press, 334 Brixton Road, London SW9. Telephone 01-733 4444.

Your Spectrum is a monthly publication.

LOTHLORIEN

WARMASTER SERIES

THE WINNING NAME IN WAR GAMES

3 NEW STRATEGY GAMES FROM THE FAMOUS WARMASTER SERIES



Johnny Reb (1861)

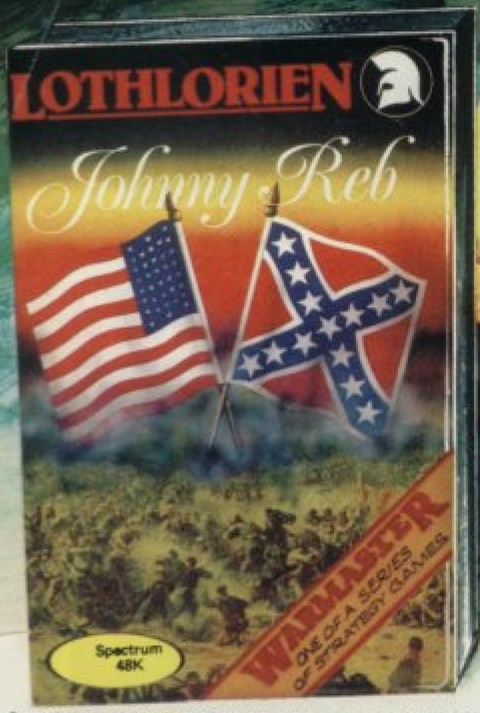
There's still plenty of fight left in the Confederate South as each side selects forces to manoeuvre on a full-graphics battle field. Use the 'save game' facility to perfect your tactics and re-write American History. 1 or 2 players.

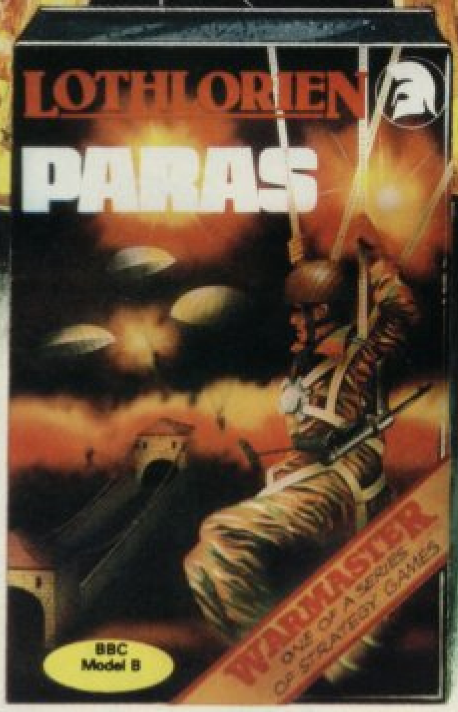
PARAS (1944)

Dramatic all-graphics reconstruction of a famous World War II Campaign. Your hand-picked force has dropped behind enemy lines to capture a strategic river crossing. 2 versions of the game with ten levels of play.

REDWEED (1995)

The Martians are coming in this superb all-graphics battle. You're in command of land forces defending London when an outbreak of semi-sentient Redweed threatens to immobilize your forces. Fifteen levels of play.







Spectrum 48K £5.50. BBC-B £6.95 and Oric 48K £6.95

Spectrum 48K £5.50.

New-Spectrum 48K £5.95.

ORDER NOW FOR CHRISTMAS

Two new two-player titles.

DREADNOUGHTS Suspense-filled high seas drama as the pride of the German Grand Fleet is locked in battle with the Royal Navy. Secret orders, battle graphics, torpedoes, guns etc.

CONFRONTATION Build-on battle game gives a basic format for any modern, non-nuclear conflict. Master tape contains modern European scenario and you can create others of your choice using both air and land forces.

More superb strategy games.

WARMASTER TITLES	
Warlord	48K Spectrum/Dragon 32 Tandy Color 32K/ZX 81 16K
Roman Empire	16K Spectrum/BBC-B/Dragon 32 Tandy Color 32K/ZX 81 16K Atari 400/800 48K
Peloponnesian War	ZX 81 16K
Battlezone 2000	BBC-B

OTHER GAMES	
Tyrant of Athens	16K Spectrum/Dragon 32/Tandy Color 32K/ZX 81 16K
Samurai Warrior	16K Spectrum/Dragon 32/ZX 81 16K
Privateer	48K Spectrum/ZX 81 16K combined cassette £4.50

Prices: ZX81 **£4.50** Spectrum **£5.50** (except Paras) **£5.95** Dragon 32/BBC-B **£6.95** Tandy **£7.95** Atari **£12.50**

Lothlorien games are available from most leading software specialists or direct by post at no extra cost. Why not really put your computer through its paces today?

NEW RED BARON—great tactical WW1 aerial combat game.

LOTHLORIEN

the mind stretcher

DREADNOUGHTS	Spectrum 48K	£5.95	☐
CONFRONTATION	Spectrum 48K	£7.95	☐
RED BARON	Spectrum 48K	£5.95	☐
REDWEED	Spectrum 48K	£5.50	☐
PARAS	Spectrum 48K	£5.95	☐
	BBC-B	£6.95	☐
JOHNNY REB	Spectrum 48K	£5.50	☐
	BBC-B	£6.95	☐
	Oric 48K	£6.95	☐

Others (please specify) ______

I enclose a cheque/PO for £____ made payable to M. C. Lothlorien

Please debit my Access A/C No. ____ /____ /____ /____

SIGNED ______

ADDRESS ______

Y.S.

Send to: M C Lothlorien, Dept. , 56a Park Lane, Poynton, Cheshire SK12 1AE. Telephone Poynton 876642

HISOFT PASCAL DEVPAC

Quality System Software

HISOFT PASCAL 4T

"... I haven't seen any other compiler that could match Hisoft's Pascal" ... *Using the Spectrum Micro — Autumn 1983.*
"This is a very impressive product ... of benefit to any Spectrum programmer" ... *David Bolton ZX COMPUTING Aug/Sept 1983.*

Just two comments from full length reviews of our powerful and virtually full implementation of Standard Pascal. The advantages of using the Pascal language are many; fast, self-documenting and truly structured programming — many schools and universities insist on a good knowledge and understanding of Pascal for their Computer Studies courses.

Now you can learn and use Pascal on your home micro; Hisoft Pascal is available for a wide range of home computers, including the 48K Spectrum. Typically, compiled programs run 40 times faster than their interpreted BASIC equivalent — and sometimes up to 1,000 times faster!

Hisoft Pascal supports all the Standard Pascal data structures and statements: INTEGERs, REALs, CHARacters, ARRAYs, SETs, RECORDs, POINTERs, Enumerated types, FOR ... DO, REPEAT ... UNTIL, WHILE ... DO, CASE ... OF and many pre-defined Procedures and Functions (e.g. SQRT, TAN, SIN, COS, INLINE, PEEK, POKE etc. etc.) — it is **not** a Tiny Pascal but a powerful educational and development tool which allows the user to develop true high-level language skills while attaining execution speed close to that of machine-code.

Hisoft Pascal for the ZX Spectrum now comes complete with ZX Microdrive support (anything you can do with tape you can do with Microdrive) and a Turtle Graphics package which allows easy creation of complex graphic programs.

HISOFT DEVPAC 3

"... DEVPAC is most highly recommended. The documentation is first class" ... *Your Computer — May 1983.*
"... it is impossible to recommend any other development package for the Spectrum over DEVPAC." ... *Adam Denning — ECM April 1984.*

Need we say more? DEVPAC 3 is a very powerful assembler, editor and disassembler/debugger. Forget about buying lots of separate packages for your machine-code development — it's all in DEVPAC 3. A full Z80 assembler with conditional assembly, all standard directives, full arithmetic (*, /, +, −, OR, XOR, AND, MOD), many assembler commands for controlling object code generation, listing, printing etc., full line editor with advanced facilities like search/replace string, 'include' facility that lets you assemble from tape (or Microdrive) without taking up memory for your source text **AND** a superb disassembler/debugger with 'front panel' display of Z80 registers and flags with commands like search for string of bytes, copy memory, disassemble memory (to an assembler textfile, if you wish), list memory in ASCII, set multiple breakpoints, and, above all, single-step through machine code programs, one instruction at a time, **EVEN IN ROM**!

DEVPAC comes complete with a 45 page user manual.

On the ZX Spectrum, DEVPAC fully supports the ZX Microdrive, allowing assembly from microdrive, saving of text to and from Microdrive and saving of object code to Microdrive — DEVPAC still fully supports tape as well.

Prices:

Hisoft Pascal 4T (ZX SPECTRUM) £25 inclusive
(NewBrain, SHARP MZ700 etc) £35 plus VAT
Hisoft Pascal 4D, many disk formats £46 inc.

Hisoft DEVPAC 3 (ZX SPECTRUM) £14 inclusive
(NewBrain) £25 inclusive

HISOFT
13 Gooseacre, Cheddington
Leighton Buzzard, Beds. LU7 0SR
Tel: (0296) 668995

*and others



It is impossible to tell you everything about the 50 games on CASSETTE - 50 but they include many types such as maze, arcade, missile, tactical and logic games, to suit most tastes in computer game playing. CASSETTE - 50 will appeal to people of all ages and the games will provide many hours of entertainment for all the family at a fraction of the cost of other computer games.

EXPRESS DELIVERY - ORDER NOW

Name ____________
Address ____________
____________ Post Code ____________
Country ____________

Dealers & Stockists enquiries welcome.

Please send me by return of post, Cassette 50 at £9.95 per tape. I enclose a cheque/postal order for
£ ______ made payable to Cascade Games Ltd.
Please debit my [Access] No. ______

SPECTRUM	☐	ORIC-1	☐	ZX 81	☐	VIC 20	☐
BBC A/B	☐	DRAGON	☐	ATARI	☐	APPLE	☐ ☑

cascade

Cascade Games Ltd.,
Suite 4, 1-3 Haywra Crescent, Harrogate,
North Yorkshire, HG1 5BG. England.
Telephone: (0423) 504526.

YS2/84

GOLD COLLECTION ■ GOLD COLLECTION ■ GOLD COLLECTION ■ GOLD COLLECTION ■

FRONTLINES

PACMAN — THE ORIGINAL

There've been any number of *Pacman* variations for the Spectrum, but all have suffered from one obvious fault — none has been allowed to stay true to the original arcade version. With Atari holding on tight to the *Pacman* copyright, anyone producing a game in the same vein is forced to make it sufficiently different — on pain of writ.

At the Earl's Court Computer Fair in the summer of last year, DJL Software (remember that excellent *Froggy* program?) had on show a true *Pacman* program — albeit an unfinished version. A notice above the screen said, "Watch our advertisements for the release of this excellent program". We did, but nothing happened!

So, at the Christmas ZX Microfair, the question was put to DJL Software: what's happened to the *Pacman* program? "Ah well, Atari will now be marketing it." So does this mean the program will be mega-expensive? "Ah well, £14.95", came the sheepish reply.

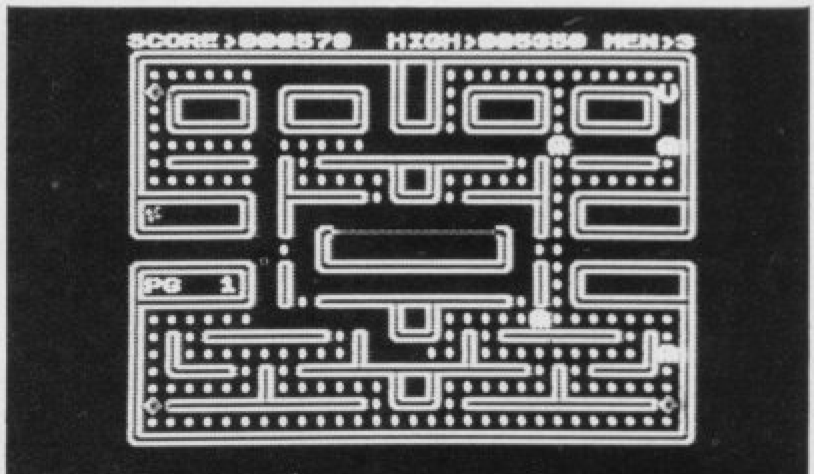
However, at last a copy fell into our grubby little paws — see the screen shot. It was a pre-release cassette, without instructions and bearing a sticky address label. It loaded quickly and worked happily with the AGF, Protek and Kempston joystick interfaces; sadly there's no provision for Interface 2, but this omission may well be rectified before the game appears officially in the shops under the Atarisoft label.

Once the type of control has been selected, the game starts with a press of the 'S' key. A very complex maze appears, that's almost the same as the one in the arcade version; the movement of the characters is smooth. The jaws of the Pacman move up and down as it eats its way around the maze and the sound — also a copy of the original — is constant throughout. The ghosts appear to have been programmed to trap the Pacman rather than just follow it! They never get stuck in the maze or follow a set path. And after Pacman has travelled through a power pill, the wraiths split up and move in all directions, making them very hard to catch.

Fruits appear on the screen at various intervals during the game, but they don't hang around for very long. And as you progress through the screens, the effect of the power pills decreases while the speed of the ghosts increases, making the game even harder.

Power Pac'ed Prices

This *is* the ultimate version of *Pacman* that many people have been waiting for — yes, it's 100 per cent true to the original. However, at £14.95, it can only be described as grossly overpriced and therefore unlikely to sell in large quantities. As we revealed in our last issue, Atari also has plans to release other arcade games under the Atarisoft label, for most home computers. These are expected to include *Moon Patrol, Defender, Ms Pacman* and *Centipede*. But don't hold your breath waiting for Atari to revise its prices before release. If they haven't learnt before, they're unlikely to learn now. **GM**



Pacman from Atarisoft. It's the real McCoy

YOUR FULLER RUMOURS

"If you've heard any rumours about Fuller, don't listen to them" — so asserts Neil Roberts of Fuller, the hardware people from Liverpool.

Rumours . . . what rumours? Well, first of all, there were stories of the fire in the Sweeting Street headquarters which destroyed much of the company's stock — which did nothing to help Fuller's struggle to meet the predicted Christmas rush for Spectrum hardware.

Then there was the teething problem with the keyswitches for the keyboards. Facing an acute shortage, one of the Fuller crew was forced to brave the wilds of Europe — eventually finding the switches he was looking for in Belgium. As a result, production of the keyboards is now well under way again, and has been since last December.



Rumours abound, but Fuller's in full swing

But that's not been the only problem. Plagued by more chip scarcity (speech chips, this time) Fuller was forced to re-think one of its Christmas ideas — to re-package the Master unit and joysticks along with four pieces of software. There just weren't enough speech chips around to make the units necessary to meet the orders that were coming in — exit one very good idea. Said Neil Roberts, "At first we got the impression that some other firm had been buying up all the speech chips. But it could have been some big dealer from the States who took them over for resale — speech chips out there are much more expensive".

Futher details from Fuller's new offices at 71 Dale Street, Liverpool 1; phone 051-236 6109. And for those who've sat on this line for weeks waiting for anything but the engaged signal, the good news is that Fuller has now splashed out on more lines. **RM**

MEET PETE THE TROUBLE SHOOTER

Having problems with a program listed in *YS*? Looking for advice on hardware and software? Or perhaps you just need the address of a company you've been trying to get hold of for weeks. Worry no longer because our unique *Spectrum Help* service will be open on Wednesdays and Fridays to sort out your problems.

On the end of the phone will be Pete Shaw, programmer, author and now agony uncle for all things Spectrumy; he's just dying to help you with your queries.

The line will be open from 10am and will close at 5.30pm (with a break between 1pm and 2pm for lunch). The number to ring is 01-631 1433.

THE GENERATION GAME

As part of a batch of educational programs announced at the end of last year to correspond with the Compec exhibition in London, Sinclair Research launched a package that includes a 'primer' on the language, Micro Prolog, together with a cassette containing the interpreter and a few example programs.

Prolog is the language that the Japanese have selected as a major component in their ambitious project for a new breed of computers called the Fifth Generation — planned for the next decade. And considering the Japanese designers are hoping to build computers that can emulate human reasoning powers, it's hardly surprising that the programming language they have chosen is a few light years away from homely old Basic.

The name Prolog is derived from the phrase 'programming in logic'. The language was originally developed at the University of Marseilles about ten years ago, and if you thought that all the programming you had done so far was 'logical' then you'd be wrong. As the authors of the primer point out, languages like Basic are 'imperative'. "Programs in these languages mostly comprise commands which specify actions to be performed."

Why else would you want to program a computer, you might ask? Well, again the answer is to do with changing the way we look at computers. So far they've have been used to enable people to 'do' things, rather than to help them think about things. All this talk of Fifth Generation, logic programming and the rest, is central to a change in our view of what computers can be used for.

Sinclair Research's version of Prolog in fact has a very respectable pedigree, based as it is on work done at Imperial College in conjunction with a project to teach primary school pupils how to program in logic. Having received no previous exposure to the deadly Basic (or, even worse, Cobol), these children will find it a lot easier to follow the arguments puts forward in the Prolog book than many Spectrum owners.

Programming in Prolog really is totally different from Basic. For a start the distinction between program and data is non-existent. Prolog programs are written by establishing a series of relationships composed of data and 'instructions'. Unlike Basic, you don't write out a series of commands and then execute them with a RUN instruction. You sort of make it up as you go along.

The example in the primer builds up a database of what might be called a typical family. So, you get things like: Henry father-of Henry-2 Mary mother-of Henry-2, and so on.

Having entered definitions of this type, you can then perform manipulations on the database — for example, which (x : x father-of Henry-2). The answer in this case would be : Henry. Sounds pretty useless doesn't it? I mean, you knew that Henry was the father-of Henry-2 in the first place — otherwise you'd not have been able to define the relationship.

There is, of course, a lot more to it than this. You can build up lists of facts and make them subordinate to other lists and then perform some fairly complex logical manipulations on them. Another example in the primer is that of a parts explosion of a bicycle and this illustrates how the language can be used more practically.

The major criticism of the Sinclair Research implementation is that you have to 'type' everything in (no using keywords as per Basic) which as we all know can be a bit wearing on the old fingertips. There doesn't seem to be any way of using the printer either — which eliminates many of the possible practical uses of Micro Prolog as a database processing tool. It also means that you cannot print your programs out. (If there *is* a way of using the printer, apologies to Sinclair Research and could they please tell me which page in the manual it's on.)

Prolog is a language that tries to relate computers to the human language rather than to strings of numbers and procedures the way that Basic does. If you're looking for new ways to build exciting graphics or make funny noises, then forget it. If, on the other hand, you want to see what the Japanese are planning and don't mind lots of typing, then it's probably worth a throw. *PM*

ANT ATTACK COMPETITION STOP PRESS

APRES LE DELUGE

Not knowing quite what to expect, the *Ant Attack* competition answerphone was switched on as the third blip of the Speaking Clock died away. And within microseconds, the phone leapt into life with our first call from Gloucester — unfortunately with the wrong answer. The second call, from Michael Kipling of London E4, burst through just as soon as the first call finished — this time with the correct locations, as well as full details of the 'AMMO' box hidden out in the desert.

What can we say? Apparently there were some 10,000 phone calls to our hotline — which according to a sorry-sounding BT engineer, almost collapsed London's 636 telephone exchange. Apologies to all those of you who didn't manage to get through and to British Telecom who had to deal with a very irate section of London's business community.

But congrats to Michael — who'll be on the receiving end of over £7000-worth of Spectrum gear. The next competition we run will be to guess what on earth he's going to do with it all!

DIAGONAL SOUNDS

New from Rainbow Electronics is a programmable joystick interface with a difference. This one actually includes a 250mW amplifier to give Speccy's feeble sound output a kick up the rear.

The interface plugs into the expansion port and will operate with any Atari-compatible joystick. And in addition to allowing the user to program vertical and horizontal movement, it'll also permit any corner position to be defined at any key, making the interface suitable for games which need to be controlled diagonally.

Early production models deactivated the keyboard — preventing its use in games calling for both joystick and keyboard control. This has since been remedied, and anyone who sends back the faulty version will have it replaced, free of charge.

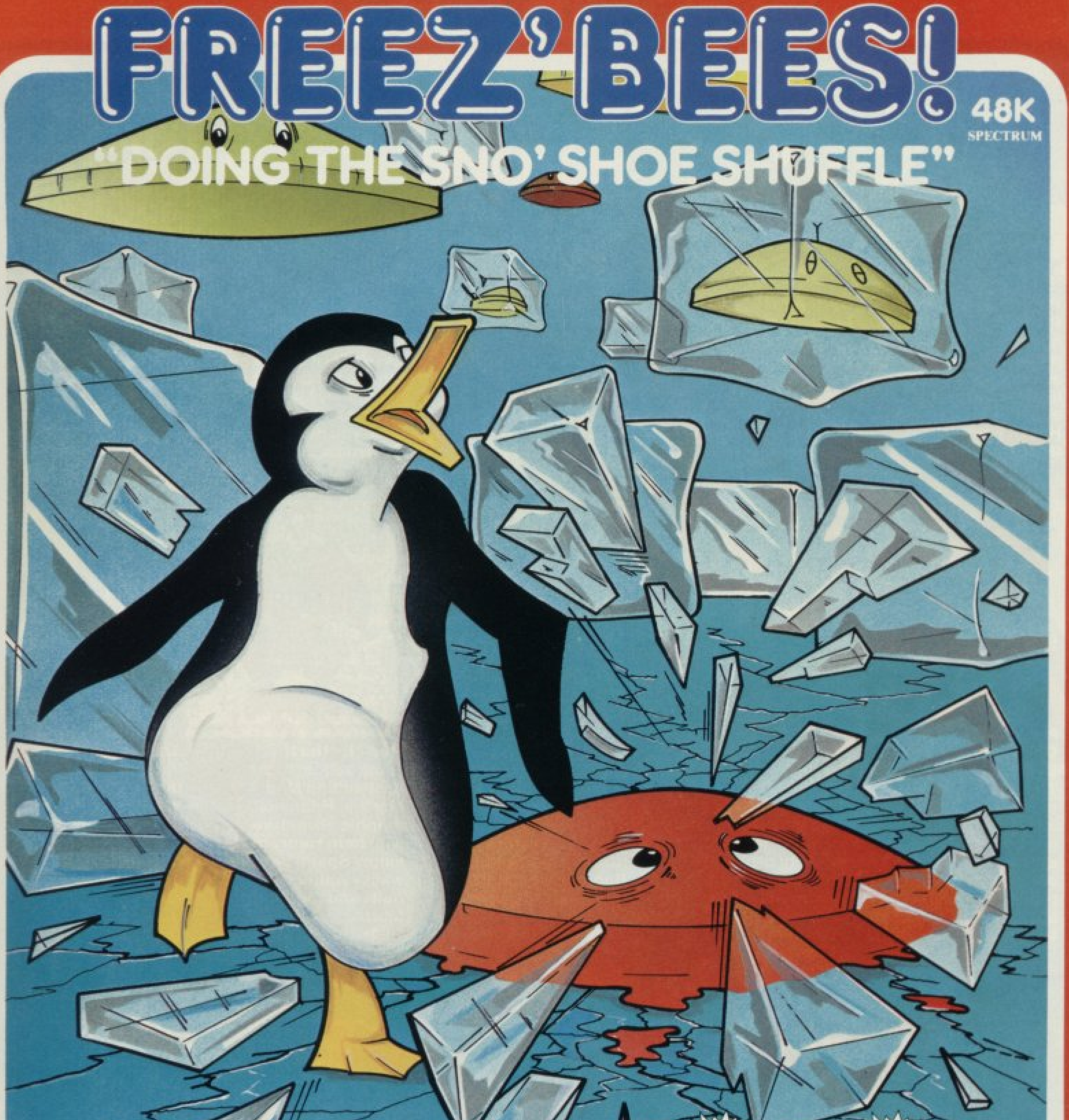
Price of the interface is £24, plus £1 p&p. You'll get it from Rainbow Electronics, Glebe House, South Leigh, Witney, Oxfordshire.

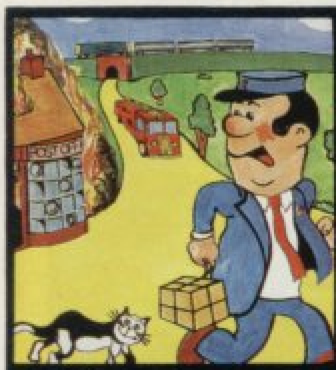


CHESS CHALLENGE WINNERS

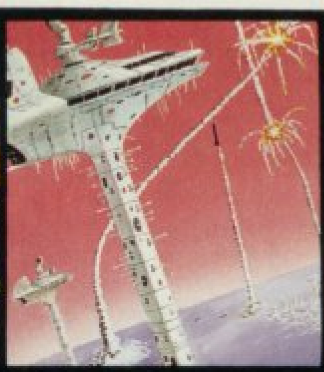
If you can remember as far back as the first issue, then you'll probably recall Professor Brainstawm's program challenge. The idea was to write a program which solved a little chess problem in Basic only.

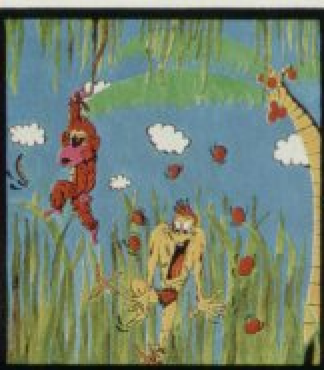
Well, we had quite a number of entries which were whittled down to three by our panel of judges. From here we left it up to our venerable Ed to decide, and the prize of a 48K Spectrum, ZX Printer, Microdrive, and Interfaces 1 and 2 will finally wing its way to Mr R Frost of Disley, Stockport. The two runners up, Mr B Partridge of London W13 and Mr P Robbins of Herts, both receive a software consolation prize for their efforts.

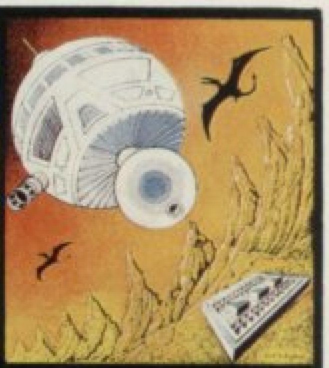

FREEZ'BEES!
48K
SPECTRUM
"DOING THE SNO' SHOE SHUFFLE"
NEW
NEW
SILVER SOFT
BRAIN DAMAGE
ARMAGEDDON
EXTERMINATOR
MISSION IMPOSSIBLE
ANY SPECTRUM
SAM SPADE
48K SPECTRUM
ROBOT RIOT











Pat the Postman
Really original. All Pat has to do is collect parcels whilst avoiding obstacles – like cars, fires, trains, etc. Skill level and Hall of Fame.
48K Spectrum. £6.95

Deffendar
Earth is under attack from mutant aliens in an accurate implementation of a top arcade game – and one of the most difficult to survive in!
48K Spectrum. £6.95

Cruise Attack
Save the city from annihilation in this wonderful version of an arcade favourite. Nine skill levels, bonus points for attack ships.
48K Spectrum. £6.95

Nanas
Catch the bananas, miss the coconuts. Easy? Try it and see. Simple in concept, hilariously entertaining – it'll drive you bananas.
16K Spectrum. £5.95

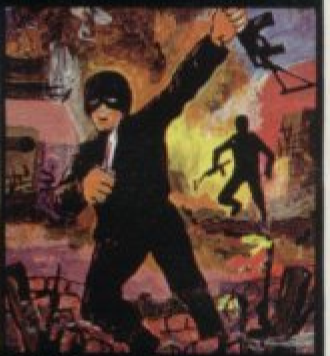
Timequest
A warp-space accident spreads your capsule across time. As you land in each new era, fresh perils face you. A very skillful graphic adventure
48K Spectrum. £6.95











One Hundred & Eighty
That famous shout tells you what it's all about! A good implementation of a difficult game – doubles, trebles, twenty-five and bull all possible.
48K Spectrum. £6.95

Land of Sagan
A quest to find the long lost Staff of Health. Is it in one of the castles? Or in the Tower? There are plenty of problems – and plenty of opponents – in this great graphic adventure.
48K Spectrum. £6.95

Creepy Crawler
An authentic version of one of the most addictive arcade games devised. All the usual features (Centipede, Spider, Bug, etc) with full use of Spectrum graphics and sound.
16K Spectrum. £5.95

Mad Martha II
Great sequel to the best-selling Mad Martha. Hilarious graphic adventure, with hero Henry in sunny Spain, beset by wacky waiters, mad bulls and the wrathful Martha.
48K Spectrum. £6.95

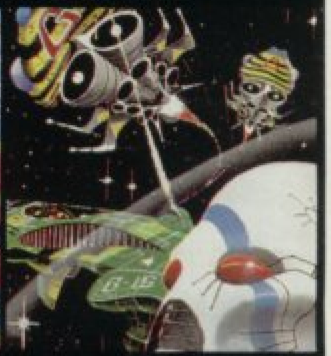
SAS Assault
Your mission – rescue the Russian ambassador from terrorist kidnappers, before the Kremlin declares war. Loads in two parts – your rating in the second depends on your skill in the first.
48K Spectrum. £6.95

WATCH OUT FOR THESE GREAT GAMES FROM MIKRO-GEN









Drakmaze
Find your way round Dracula's domain. A game to get your teeth into!
48K Spectrum. £6.95

Laserwarp
Invaders, Aliens, this game has the lot – and you have to survive to destroy the Master! Simple controls, far from simple task.
48K Spectrum. £6.95

Knockout
No aliens, lasers, invaders or rockets – just simple but absorbing fun which up to 4 players can enjoy.
48K Spectrum. £6.95

Galakzions
Watch out as the Galakzions break formation to attack in such force that no mere human can survive!
16K Spectrum £5.95

See them at Lightning, PCS, Tiger, Co-op, Menzies, W H Smith & other leading retailers

PHONE YOUR VISA OR ACCESS NUMBER

Other great new Mikro-Gen programs coming soon – watch out for them!

Please make cheques/PO payable to 'Mikro-Gen' and add 40p post & packing per order.

MIKRO-GEN

Mikro-Gen, 1 Devonshire Cottages, London Rd, Bracknell RG12 2TQ   Tel: 0344 27317

HOLD THE FRONT PAGE!

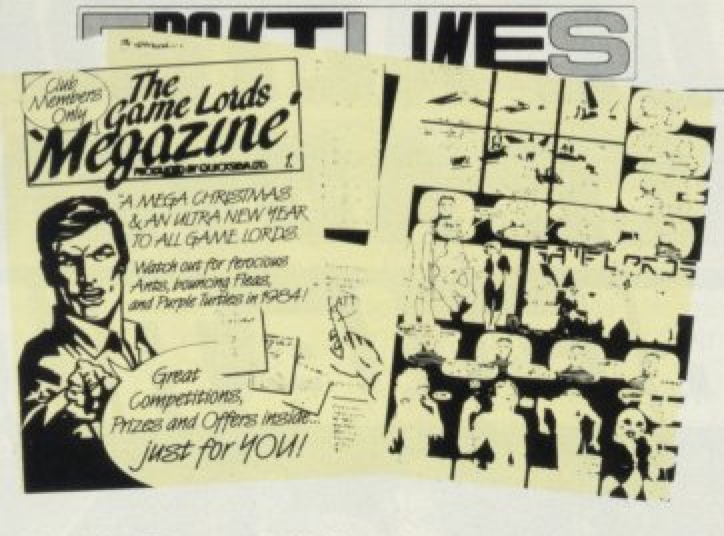
Asked to pick any one software house that was light years ahead of the rest, many would go for Quicksilva — not only for its wide range of software, but also for the company's love of the wilds of space. Read any Quicksilva literature, be it games descriptions or adverts, and always they'll be in 'Sci Fi' speak.

And after all the playing around with cassette sleeves and advertising, Quicksilva has finally switched to print proper — with its new *Game Lords* magazine. In charge of the new venture is Caroline Hayon, now to be known as the Chief Game Lord Droid, and it's in fact the official newsletter of the Game Lords Club.

The magazine, eight pages of A4 printed in black and white, contains all the details of the offers available from the club. Membership gets you money off all Quicksilva's software, as well as the chance to fill your cupboards with T-shirts and badges proclaiming Quicksilva to be the best thing that ever happened to the known universe

Inside *Game Lords* there's a feature on the history of the Quicksilva company (albeit with a slight degree of poetic licence) and there are all sorts of competitions and prizes to be won. The centre spread of the magazine takes the form of a real live comic strip about . . . yes, you've guessed it . . . the Game Lords. It's well-drawn by Chris Brasted.

Membership to the club can be made via Caroline Hayon — sorry, Chief Game Lord Droid — at Quicksilva Ltd, Palmerston House, 13 Palmerston Road, Southampton SO1 1LL. And all it costs is a one pound 'greenie' (or one of those slot machine tokens). *YS* will be following the club's fortunes with great interest. **RM**



A 'greenie' will buy you all this and more

FLYING PIGS DEPARTMENT

The recent LET show at the Heathrow Penta Hotel found Phoenix Software's managing director, Gerald Rose, positively jumping for joy and regaling all and sundry with news of a piece of software that would have a dramatic effect on the Commodore 64 market — not to mention making him a mint in the process.

Mr Rose had just got word from one of his freelance programmers — an IBM whizz kid — that the wonder-program was finally finished. It would, he said, enable Commodore 64 owners to make use of the wealth of Spectrum software — a tricky bit of software that would allow it to run on their own machines. Certainly, it sounded an extremely sophisticated piece of programming, and something that might make him a touch unpopular with some of his competitors. Mr Rose was unmoved. He claimed the prototype had been "tested on at least 10 packages picked at random from the shelf, and had worked perfectly in every case". The only thing was that his programmer "needed the weekend to finish it off". As soon as a production copy became available it would be despatched to *YS*.

Days passed, and weeks passed . . . nothing more was heard from Mr Rose.

Just in case Phoenix had forgotten about us, we phoned its offices and were told by a spokesperson (probably the office junior) "The program's got a bug in it, and doesn't work. We've had to send it back to the programmer for debugging. And until it's been successfully sorted out, we're saying nothing to the press". Ah well, such is the stuff of which dreams are made. **RS**

E-TYPE INTERFACE

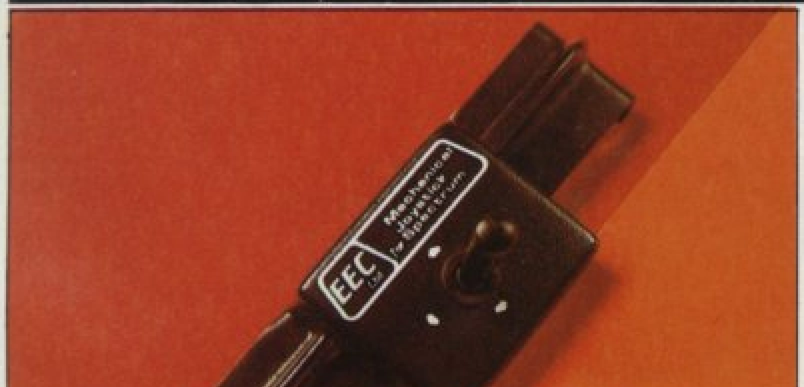
Kempston Micro Electronics, probably best known for its joystick interface, has now improved its popular Centronics Interface. Called the 'Model E', it contains an EPROM which has all the necessary software to drive Centronics printers. This is a great advantage when you consider that most other printer interfaces require software to be loaded from tape. In fact, with the exception of the Fuller Printerface, this is the only Centronics interface to contain its software on EPROM.

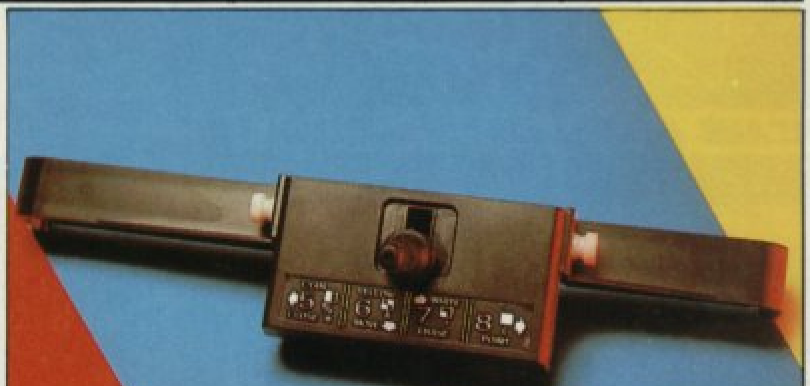
Included in the software is a copy routine for dot-matrix printers. This works using the actual COPY command rather than a RANDOMISE USR, a great advantage when you forget where the COPY routine is stored.

The software on Eprom has been very well put together from the user's point of view. Most notably, let's look at the COPY handling routine. If you enter COPY: REM ?, the software puts a status table on the screen, displaying the current state of the interface. You can then change it to suit your particular printer.

Although at £55 it's a little more expensive than most, the unique features would seem to cover the extra cost. More details from Kempston Micro Electronics, Unit 30, Singer Way, Woburn Road Industrial Estate, Kempston, Bedford MK42 7AF. Or phone 0234 856633. **PS**

LOOK MUM, NO WIRES!





Standing lonely amidst a sea of joystick manufacturers all claiming their product's compatibility with any games program on the market — thanks to the immense wizardry of their technical department — here are two joysticks who've ignored all the electrical bits and pieces. Both devices are designed to be clamped atop the Spectrum in such a way as to depress the '5', '6', '7' and '8' keys when the 'stick' is moved in a particular direction. Compatible with all games using the direction keys for movement (or those allowing you to define the keys), both joysticks are priced at £9.95 — which is a good deal cheaper than their electronic counterparts. For further info, bell EEC Ltd on 0753 885401 or Grant Designs Ltd on 0603 870852 and check 'em out yourself.

WHEN THE LEVY BREAKS

The illegal copying of recorded material — records, music cassettes, video cassettes, computer tapes and so on — as well as the recording of broadcast material, has become so widespread that the industries involved are once more calling for a levy on all blank audio and video recording tape. This could well lead to a staggering 100 per cent increase in price, a hike of which would not only be felt by those involved in illegal copying, but also by consumers with a legitimate use. Purchasers will in effect be penalised for a crime they haven't committed.

The group in favour of these levies is known as the International Federation of Phonogram and Videogram Producers (IFPI), and it's made up of members of the record, film and television industries who, not surprisingly, are precisely those people who would most profit by the action. Their argument is simple and straightforward. They claim that piracy and bootlegging is losing them many millions of pounds each year, and is responsible for the decline of these industries. But *YS* believes there is an important distinction to be made here, and it asks if it's fair and realistic to blame the substantial loss of profit suffered by IFPI members on the person who makes an occasional recording from the radio or TV for his or her own personal use? Surely, it's the highly organised pirating for gain which is really to blame, and a reform in the copyright act calling for tougher laws with stiffer penalties for those found guilty would not only help to stem this practice — it would also be a far better way of dealing with the 'real' culprits.

Fighting against the IFPI's bid is the Tape Manufacturers Group, an organisation which consists of the seven leading suppliers of audio and video tape in Britain — Agfa, BASF, 3M and Maxell, Memorex, Sony and TDK. Its aim is to make all users of blank tape aware that the tape levy issue is rapidly coming to a head throughout the European Community; if the proposal is to be blocked, then action must be taken now.

But how does all this affect the personal computer industry? Well, according to the TMG, it looks as though both software houses and consumers alike will have to suffer the consequences of such a levy, without receiving any of the benefits. Certainly, it's quite unlikely the software houses would receive any of the revenue by way of compensation for having their software pirated.

Asked whether he thought the levy would indirectly put up the price of cassette-based software, a TMG spokesman said "I imagine it must. If the software companies are going to be charged more for their blank cassettes, they'll probably pass the cost on to the consumer". And he went on, "I think this levy will stifle the sales of computer tape. It's rough justice for the user and, rather than prevent illegal copying, it could signal the start of home taping on a massive scale. People are going to feel that they've been made to pay for the privilege, which means they'll use it".

But what position does the software industry take? Well, a letter from Rod Cousins, Vice Chairman of the Guild Of Software Houses (GOSH), urges magazine editors to "act in a creditable manner" by "adopting a responsible attitude in the acceptance of advertisements which may be considered against the best interests of the industry". The letter also states that GOSH "are active in seeking measures to restrict the illicit copying of computer programs".

And so, while the debate continues, the time is fast approaching when Westminster MPs must decide whether to vote for or against tape levies. But if parliamentarians decide they don't want them, the European Parliament could force Britain to accept them. **RS**

A CASE STUDY



For those aspiring to the 'executive' look (but growing old awaiting the precious QL), why not think about a case that incorporates a console for your Spectrum, ZX Printer, power pack and tape recorder. For those who've already managed to wrap hands around an Interface 1 unit, there's a separate console for storing said device plus up to three Microdrives. The price of the non-gold encrusted case is also in the 'executive' range — £47.45. Those feeling flush should contact Treetop Designs, 61 Widmore Road, Bromley, Kent BR1 3AA.

SULIS SPELLS IT OUT

Although the DoI has been actively encouraging the use of the microcomputer in schools, it's been left to independent manufacturers to fulfil the need for software that has some 'real' educational value. And to this end, Sulis Software has put together a range of spelling and vocabulary programs which should guarantee to get the little urchins' grey matter all powered up.

All programs in the Sulis educational range are priced at £9.95. For more details contact Sulis Software Ltd, 4 Church Street, Abbey Green, Bath BA1 1NL or telephone 0225 61929. **RS**

REPEL BOARDERS

It's not easy to protect a Basic program on a Spectrum. Because the machine is so open, a potential software pirate can BREAK and LIST a Basic program with ease. One way is to adopt a programming style that is so convuluted that no one could possibly understand it — the usual fashion amongst some of the less experienced games writers!

A more professional approach, however, is to mess around with the system variables using POKE statements. These are defined on page 173 of the Spectrum manual published by Sinclair Research. On the same page is the basic equipment for playing about with the system variables. Before tampering, make sure you read the notes that Sinclair Research has so kindly provided on which items you can POKE and which you can't. It's not a good idea to meddle with the various flags that control the interpretation of your Basic program, for example.

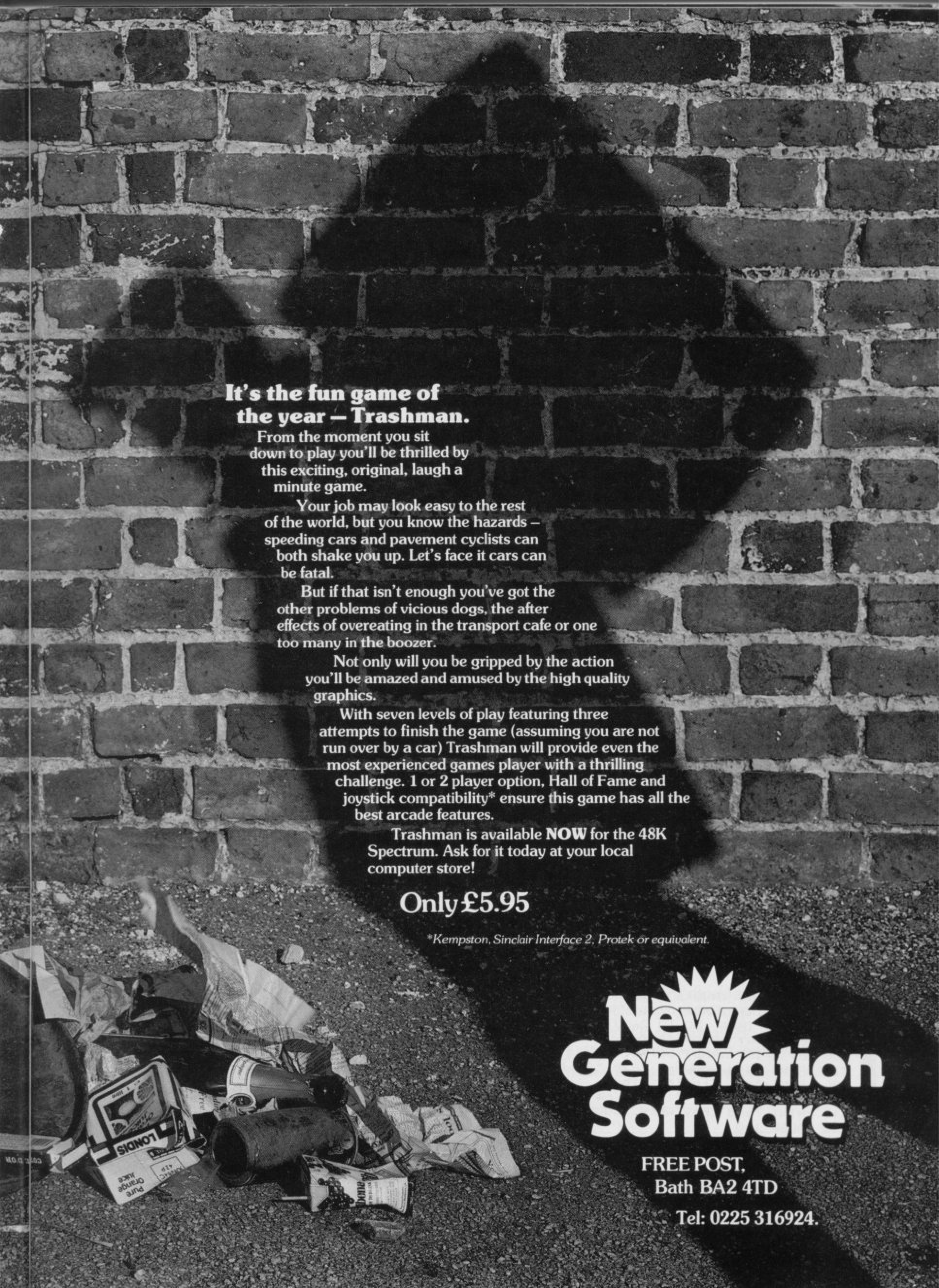
A straightforward way of making a Basic program unreadable is to redefine the character set with one of your own and POKE the appropriate system variable to point to your coded character set. The variable in this case is found at location 23606. Being a two-byte field, this requires two POKEs to get your new character area address in. The technique for doing this is also on page 173 of the Spectrum manual.

However, for those who have lost their manual (or those who cannot be bothered to turn the pages over) it goes like this:

```
POKE 23606,address–256*
   INT(address/256)
POKE 23607, INT
   (address/256)
```

The address is that of your new character definition area. If you need to return to the standard Spectrum character set, you do the same POKEs using 15360 as the address. This would obviously apply if you wanted to list the program yourself. This method of protecting the listing has an added bonus in that you can use the character redefinition table to define your own characters — should you run out of user-definable characters in the standard set.


Trashman
GSH
Guild of Software Houses
New Generation products are sold according to their terms of trade and conditions of sale.


It's the fun game of the year – Trashman.
From the moment you sit down to play you'll be thrilled by this exciting, original, laugh a minute game.
Your job may look easy to the rest of the world, but you know the hazards – speeding cars and pavement cyclists can both shake you up. Let's face it cars can be fatal.
But if that isn't enough you've got the other problems of vicious dogs, the after effects of overeating in the transport cafe or one too many in the boozer.
Not only will you be gripped by the action you'll be amazed and amused by the high quality graphics.
With seven levels of play featuring three attempts to finish the game (assuming you are not run over by a car) Trashman will provide even the most experienced games player with a thrilling challenge. 1 or 2 player option, Hall of Fame and joystick compatibility* ensure this game has all the best arcade features.
Trashman is available NOW for the 48K Spectrum. Ask for it today at your local computer store!
Only £5.95
*Kempston, Sinclair Interface 2, Protek or equivalent.
New Generation Software
FREE POST, Bath BA2 4TD
Tel: 0225 316924.
LONDIS
Pure Orange Juice

JE PARLE SPECCY

Silversoft Ltd has announced the release of Dr Michael Gruneberg's *Linkword System* for the 48K Spectrum. Designed by Gruneberg, the language learning system was first released in 1982 and received wide media acclaim the following year when Thomson Holidays — Gruneberg's first client — published figures that showed 98 per cent recall had been achieved during management training. This impressive figure led the company to implement *Linkword* in parallel with its own in-house language courses.

Silversoft claims that in little over 10 hours you will be taught a vocabulary of more than 350 words, plus a basic knowledge of grammar. The words the system teaches are obviously aimed towards the holidaymaker, although it could quite possibly be of some help to the reluctant pupil — or just about anybody who wants to learn a language without too much fuss.

The package includes two cassettes, one audio and the other containing the program itself. The audio tape is there to help you pronounce the words correctly.

When the *YS* team was let loose on the *Linkword* Spanish system, it did seem that we picked up some words a lot easier than we expected. So we asked Andre Posumentov of Silversoft why he thought the *Linkword* system was so good. With great enthusiasm he replied "In *Linkword* we have a properly structured and field tested course which gives someone a grasp of the language very quickly. *Linkword* doesn't just present the information, it *teaches* you."

As with most of the other educational packages, it's hard to say whether the computer is really necessary. Wouldn't it be just as good if you were reading it all out of a book? But the main idea behind the Gruneberg system is vivid images which you create in your own mind. There are no pictures or graphics included in the programs.

Of all the language programs currently available for the Spectrum, this one seems to come out on top. It's available in three different versions at the moment — French, Spanish and German — and Italian, Greek and Portugese are on the way. Each course costs £12.95 and should be widely available from most software shops. **PS**

NO RUBBISH HERE

New Generation Software has announced the launch of its latest game, Trashman — and the company's convinced it has a top ten best-seller on its hands. It claims "thrilling action, superb graphics and highly imaginative playing elements".

Players must collect and empty dustbins into a moving rubbish cart while avoiding a variety of hazards — like speeding cars, pavement cyclists, vicious dogs, over-eating and one too many in the pub. There are seven levels of play, which is not al that many when compared with some of the games around today, but if the end result is anything like as good as NGS's previous software then it must have credibilty.

DEBUGGIT!

```
10 REM DEBUGGIT!
20 DIM n$(10,15)
30 DIM p$(1,20)
40 DIM w$(10,60)
50 PRINT "Type in the names of ten of
yourfriends, or enemies."
60 FOR a=1 TO 10
70 INPUT n$(a)
80 PRINT 'n$(a)
90 NEXT a
100 CLS
110 PRINT "Type in the names of  ten pl
acesyou know."
120 FOR a=1 TO 10
130 INPUT p$(a)
140 PRINT 'p$(a)
150 NEXT a
160 CLS
170 PRINT "Type in ten things you like
    doing best."
180 FOR a=1 TO 1
190 INPUT w$(a)
200 PRINT 'w$(b)
210 NEXT a
220 CLS #
230 PRINT "Now for the good bit..."
240 PAUSE 100
250 CLS
260 LET f=INT (RND*10)+1
270 LET s=INT (RND*10)+1
280 IF s=f THEN  GO TO 2600
290 LET p=INT (RND*10)+1
300 LET w=INT (RND*100)+1
310 PRINT "Did you now that "'n$(f)
320 PRINT "met ";n$(f)'"in"
330 PRINT '''''''''''''''''p$(p)
340 PRINT "and they were caught "'p$(w)
350 PAUSE 100
360 PRINT AT 21,6;"Press any key"
370 PAUSE 0
380 GO TO 250
```

Well, would you believe it . . . our own lovable DH has actually managed to supply the corrections to last month's Debuggit. Unfortunately, no-one could read his scrawlings, so we had to work out the bugs for ourselves anyway! Here they are:

1. Line 80 should read INPUT a$: PRINT a$, and not the numeric variables that Fatface put in.
2. Lines 140 and 480 were transposed.
3. In lines 200 and 220, a$ should equal 'a', 'b' and 'c' instead of 'x', 'y' and 'z', respectively.
4. Lines 590, 600 and 610 were all over the shop — line 590 was where line 600 should have been; line 600 was in place of line 610; and line 610 should have been positioned at line 590.

Of course, there's always more than one way to skin a cat, or in this case, a bugged-up streak of programming incompetence called DH! So just as long as the whole thing totters through when you press RUN, consider the program de-bugged.

'OH NO, NOT AGAIN' DEPARTMENT

It had to happen — this month Dick Head insists on presenting his latest bundle of chip-shattering Specy fodder all by himself. So, on the grounds that if we give him enough rope he might just turn himself into an endless loop and disappear up his own RAM port, here goes!

"Err, hi . . . okay you guys, this is it. All you've got to do is — umm — type in the names of some of your friends, places you know, and fings you like doing. Then the program goes into some a-mazing jumbling routines, and ends up writing some first clarse poetry. Only fing is, I haven't quite got it to work properly yet . . ."

Good Lord, what a surprise! While DH is away this week, brushing up on how to defend himself against sudden vicious assaults from other members of the *YS* staff, why not cast your eyes over his latest horror story — who knows, it might even be fun. As usual, Dick promises to have the whole thing sorted out by next issue — and, of course, I believe him. Ha-ha-ha hey, what are doing with that funny jacket . . .

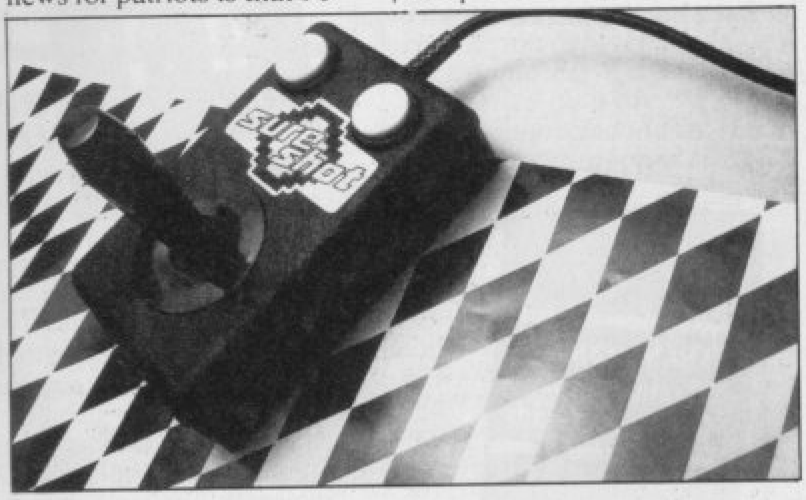

48K ZX SPECTRUM
The Knights Quest
THE FOREST
PHARAOHS TOMB
48K ZX Spectrum
LOONY ZOO
DO NOT FEED THE EARTHMAN
COLDITZ
48K ZX Spectrum

COOKRIDGE SURE SHOT

New from Cookridge Computer Supplies comes the Sure Shot joystick which, claims Cookridge's Lawton-Harris is "by far the most reliable and well tested on the market and a variation of one that has so far sold in its hundreds of thousands to arcades". With all that expertise behind it, it's hardly surprising the company has decided to adapt the device for home computer use.

Its base is a rigid steel chassis on to which all the components are fixed. Good news for patriots is that 90 per cent of these widgets are claimed to be of British manufacture, and the whole is housed in a case made of high impact ABS plastic. Other features include two fire buttons, a phosphor bronze bearing and nylon actuator *(thank heavens for that! Ed).* All this for £15.95, the only 'extra' being a programmable interface that costs £22.95.

Both of these products can be had via mail order from Cookridge Computer Supplies, PO Box 1W9, Leeds LS1 6NT — or telephone 0532 670625.



THREE FROM RICOLL

It's said that three new add-on products from Ricoll Electronics have proved so successful that supply is only now returning to normal. The first is the company's professional-standard keyboard. Housed in a coffee-coloured case, its cream keys are guaranteed for 50×10^6 (50 million) operations — a pretty high degree of reliability. It also features a proper space bar, with an extra Shift key on the right. Supply has reportedly improved drastically, with an order to delivery time of two to three weeks.

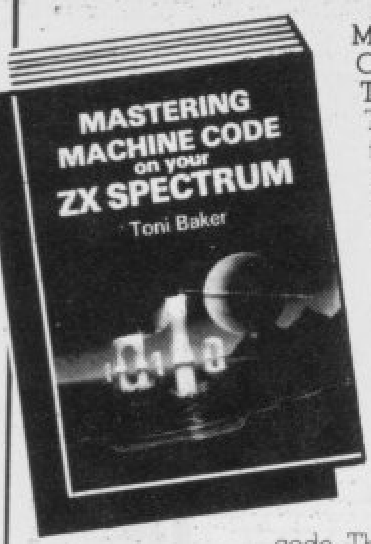
Sound boards are obviously very useful when it comes to improving the 'playability' of computer games and Ricoll's does more than most. It'll produce the kind of music and sound effects normally to be found only on full-size monophonic synthesisers. The gadget includes three oscillators (programmable from 0.01Hz to 62KHz) and gives a square wave ouput. Other notable features include a noise generator, envelope shaper, three audio channels with programmable volume, internal audio amplifier and a 16-bit I/O port. Combine all this lot with the programmable filter and you'll be synthesising many sounds that rival units will be unable to generate.

On the subject of sound 'extras', there's Ricoll's Action Replay box which plugs into the Spectrum's expansion port. It's not actually a sound synthesiser, and is used more for 'real' sound effects. Using a microphone, a sound can be input and sampled, sorted in memory and then replayed at any pitch. The most interesting feature is the device's ability to create glitch-free tape loops for producing the most amazing sound effects.

The keyboard costs £37.95 plus £2.00 p&p, while the sound generator is £39.95 plus 70p p&p. The Action Replay box which still has an order to delivery time of six to seven weeks, is expected to sell for just under £100. All equipment is available from Ricoll Electronics Ltd, 48 Southport Road, Ormskirk, Lancs. For more info, telephone them on 0695 79101.

RS

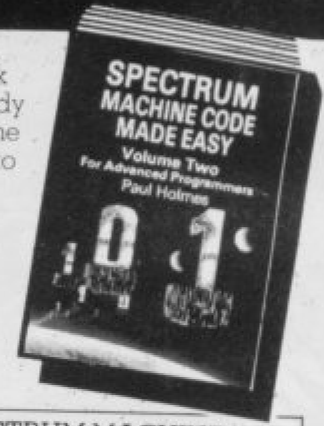
At last!
MASTER MACHINE CODE on your Spectrum

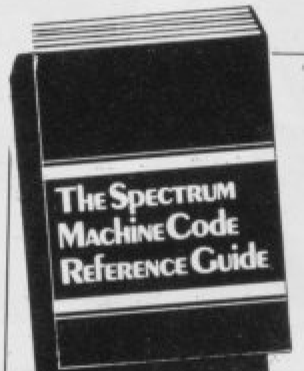


MASTERING MACHINE CODE ON YOUR ZX SPECTRUM
Toni Baker £9.95
This 315-page book is designed to teach you the essential elements of programming in machine code. Written by Toni Baker, author of the highly successful 'Mastering Machine Code on the ZX81', this new book assumes absolutely no previous knowledge of machine code whatsoever, and yet promises to take you to a level of proficiency beyond your wildest dreams. Starting with simple addition and subtraction you'll be slowly guided through the entire subject of machine code. The book explores and utilises the incredible speed of machine code, giving you real time graphics games like BREAKOUT and leads you up to a full working DRAUGHTS program. Among other useful skills you'll acquire the ability to create music in real time (impossible in BASIC) and to plot in high-resolution graphics faster than you would have believed possible.

SPECTRUM MACHINE CODE MADE EASY
Volume One — James Walsh
Volume Two — Paul Holmes
These two books provide a graduated course in machine code programming on the ZX Spectrum. Book one starts off with the basic concepts of machine code, followed by an explanation of binary maths, hexadecimal and base conversion, leading as quickly and painlessly as possible onto the rules and types of addressing the Spectrum's powerful Z80 microprocessor. Book two is designed for those who already understand the rudiments of machine code programming, and now wish to increase their skills.
Each book is just £5.95







THE SPECTRUM MACHINE CODE REFERENCE GUIDE
Microdrive, Interface 1, and ROM Disassembly
Richard Ross-Langley £4.95
This 170-page reference work for Spectrum machine code programmers contains a full disassembly of the Spectrum ROM, with details of the Microdrive and Interface 1. Features of the disassembler include: Zilog mnemonics are used, eg LD A, (HL) instead of MOV A, M; relative jumps show the signed decimal offset and the result; hex values are default and are printed without suffix; decimal values are preceded by a plus or minus sign; and some restart instructions are followed by data bytes. The absolute addresses of all system variables and several important routines have been named, using where possible the standard names shown in the manual. The chapter headings in the Microdrive/Interface 1 section of the book include the RS232 Interface; Microdrive Channel data; Local Area Network; Network Algorithms; System Variables; and a summary of functions. **This book is a must reference work for serious Spectrum machine code programmers.**

Interface Publications, Dept. S, 44-46 Earls Court Road, London W8 6EJ

Please send me the following books. I enclose £__________

- ☐ Mastering Machine Code on your ZX Spectrum — Toni Baker — **£9.95**
- ☐ Spectrum Machine Code Made Easy, Volume One for beginners — James Walsh — **£5.95**
- ☐ Spectrum Machine Code Made Easy, Volume Two for advanced programmers — Paul Holmes — **£5.95**
- ☐ The Spectrum Machine Code Reference Guide — Richard Ross-Langley — **£4.95**
- ☐ Creating Adventures on your ZX Spectrum — Peter Shaw and James Mortleman — **£4.95**
- ☐ Putting Your Spectrum to Work (includes 15 major programs, including a word-processor and spread-sheet calculations) — Chris Callender — **£4.95**
- ☐ Creating Arcade Games on the ZX Spectrum — Daniel Haywood — **£3.95**
- ☐ Programming Your ZX Spectrum — Hartnell/Jones — **£6.95**
- ☐ 60 Games and Applications for the ZX Spectrum — David Harwood — **£4.95**
- ☐ Beyond Simple BASIC — Delving Deeper into your ZX Spectrum — Dilwyn Jones — **£7.95**
- ☐ Instant Spectrum Programming (tape and book) — Tim Hartnell — **£4.95**
- ☐ 20 Simple Electronic Projects for the Spectrum — Stephen Adams — **£6.45**
- ☐ Giant Book of Computer Games — Tim Hartnell — **£7.95**

Name:..

Address:..

....

INTERFACE PUBLICATIONS
We're the experts!

All books available from computer and book stores, including W H Smiths, Menzies, and Dixons.
Trade supplied by: The Computer Bookshop, 30 Lincoln Road, Olton, Birmingham B27 6PA (021 707 7544, telex 334361).

FORUM

If you fancy 'writing on spec' — put pen to paper and tell us what's on your mind. Post letters post-haste to: Forum, Your Spectrum, 14 Rathbone Place, London W1P 1DE.

THE HEAVENS OPEN

Thank you for the copy of your latest issue of *Your Spectrum* and the *QL User* supplement.

Obviously, we welcome this sort of support for the new machine and hope we can reciprocate by getting large numbers into users hands very quickly.

With best wishes for the success of *QL User*.

Nigel Searle, Managing Director, Sinclair Research, Cambridge

Thanks also to Sir Clive himself, who wrote us a letter saying he found the mag "very interesting". Praise indeed! Ed.

SOME SPECCYS 'AVE ADAM

Regarding the *Spectrum Keyboard Buffer* article in the Jan 84 issue, my problems and comments are as follows.

Fault: I have built the buffer with the intention of using it as a joystick buffer by replacing the IC socket with a screwed connector block. Connecting up individual address lines and data lines singly, both by using a shorting link at the common line and a switch connection from a Commodore joystick, I get correct key operation.

When I connect up the common line of the joystick to A12 and the four positional switch lines to D1 to D4 inclusive, I get no reaction at all but the copyright is OK, etc, and the keyboard works if I disconnect all but one data line, then the joystick operation in that direction will operate the Spectrum.

Comment: on construction I found that in your illustration of the veroboard and connection you have linked pins 3 and 4 on the keyboard connector, ie. the 5v to the 9v. This, I think, could be very expensive for the Spectrum owner. Yes?

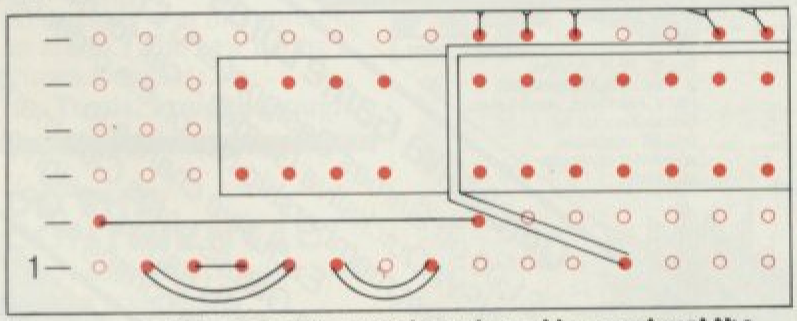
Simply as a suggestion, would it not be a good idea on any reprints to name the lines shown coming away from the 74LS245 central connections?

I would be obliged if you could solve this problem.

Mr Crasley, Basildon

Apologies! The circuit diagram given in my article was slightly wrong. Pin 14 at the end of the gate should have been labelled Pin 12. There should have been no connection between the IC socket and Pin 7. The two unlabelled lines are +5v and 0v (0v connected to Pin 7).

The new veroboard straps are shown in the diagram. Any circuit should be checked before *connecting to the Spectrum. The ICs used are standard types and pin connectors can be looked up in a component catalogue if there is any doubt. Mr Crasley did this successfully and his problem with the joysticks should be solved by adding a resistor to each data line on the socket side, connected between the data line and +5v (10K ohms should do). Stephen Adams.*



The corrected straps for the veroboard used to construct the keyboard interface as published in the article Spectrum Keyboard Buffer in YS issue one.

PLAYING HARD TO GET

I refer to the program, *Showdown*, in the *Play Power* section of your first issue.

Half-way through the program, you comment that the keyboard routine may not work on the new issue 3 Spectrums. As people typing in the program will have spent some time at the keyboard by the time they reach line 190 (and probably have no idea which issue Spectrum they have anyway!) that perhaps was not the best place to break the bad news.

However, the program *will* work with the issue 3 models if the value of the IN command is changed from 255 to 191. On other occasions, when converting programs using the IN statement for the issue 3 Spectrum, you may like to refer to the following list (the IN command values are followed by the altered values for the Issue 3 in brackets): 255 (191), 254 (190), 253 (189), 251 (187), 247 (183), 239 (175).

RP Taylor, Cheshire

OUT OF THE CLOSET...

I wonder if I could use the letters section of your magazine to make an appeal to your readers? (*Be my guest. Ed.*) I am involved with an ILEA Saturday school project in Brighton. As part of our activities I run a computer group, and although we have access to three Spectrums, we desperately need more facilities. There must be a number of people who have discarded their ZX80s and ZX81s to cupboards having bought Spectrums. Perhaps some of these people would be willing to loan or donate such unused micros, as the funding for the project is very low. If anyone can help could they please contact me.

Stephen Oxford, Senior Teacher, Highbury Fields School, Highbury Hill, London N5 1AR

WAKE UP AT THE BACK!

Quick! Print this before our MD breaks out of his cage and does something drastic! What am I talking about? Well, actually it's about the review of our ADS Centronics interface that you ran in your last edition. Your reviewer states that our manual needed reading two or three times to glean all the available info. I guess he's right, because he obviously didn't.

His parting shot at our little protege was "superb unit, but slightly let down by the provision of supporting high resolution dump software for only one range of printers".

Mmm, interesting one that. You see, when we sat down and started looking at what kind of software we should supply with the unit, we sussed out the competition and sure enough, we saw that they provided separate dump routines for each type of printer.

"Hey, what if we could produce a dump routine that runs on all printers, with the minimum of alteration?" says wily MD. "No problem" replies resident genius, result — a routine that *is* adaptable for all printers, using just a few (well documented) POKEs. Yes, *YS*, our unit will only support one range of printers — dot matrix ones. Come on you reviewers, at least read the instructions first and give us poor designers a break, eh?

Martin Flenley, Software Designer, Advanced Digital Systems Ltd, Portsmouth

IT ALL ADDS UP

I have to say that I'm amazed at the first prize in the *Ant Attack* competition.

I got out my trusty calculator and worked out that the prize is worth over £7000. With that sort of money you could buy an Apple IIe, disks, a printer, a modem and still have enough money left over for software and electricity bills.

Iain Radford, Derbyshire

Good thinking... but why on earth would you want an Apple??! Ed.

RESTEZ REGULIER

I was agreeably pleased to read, during my Christmas holidays in England, issue 1 of *Your Spectrum*.

I appreciate the versatility of the articles and games descriptions, etc.

In a phrase 'Pour un coup d'essai ce fut un coup de maitre'. (*Er, right. Ed.*)

Congratulations! However, I was surprised by the lack of a subscriptions voucher. Will you be including one in future issues?

ER Boulad, Courbevoie, France

Details of how to subscribe to Your Spectrum *are included in this issue. Ed.*

WHAT'S THE CONNECTION?

In the January issue of *Your Spectrum*, the article about the Currah MicroSpeech unit, *Talk to me, Oh*

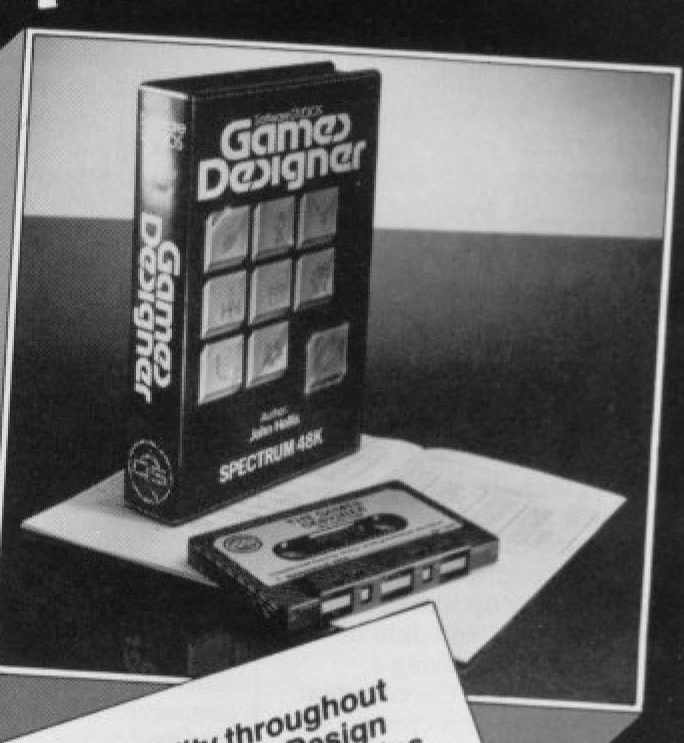
A lifetime of games in one program!

CREATE & PLAY YOUR OWN ARCADE GAMES! Simple to use

Software STUDIOS

Games Designer

By John Hollis
Author of Time Gate, Aquaplane and Meteor Storm.



Invaders format Asteroids format Defender format & Beserk format

8 GREAT GAMES INCLUDED:
Turbo Spider-Tanks a lot-Halloween-Attack of the Mutant Hamburgers Cyborg-Reflectron-Splot and Q-Bix

Smooth Arcade quality throughout using full colour sprites • Design your own fully animated Aliens, Ships, Missiles, Explosions • Full attack wave design (8 waves per game) • Moving background option • Full sound generation • Full movement control and scoring • Save and load YOUR OWN games to and from tape!

£14·95
SPECTRUM 48K
Dealer enquiries welcome.

Quicksilva Game Lords Club Members
£1.00 off Games Designer and £1.00 off Aquaplane and other NEW Quicksilva Games – call for details. Quote your secret membership No. on the order coupon.
Exclusive chance to Quicksilva Games Lords Club members to have their OWN games design published! Details coming soon!

Marketed exclusively by Quicksilva

Games Designer	£14.95
Time Gate (48K Spectrum)	£6.95
Aquaplane (48K Spectrum)	£6.95
Meteor Storm (16K Spectrum)	£4.95

I enclose cheque/P.O for________
Please send me details of the Quicksilva Game Lords Club ☐
My Game Lords Club membership number is________

Name________
Address________

Cheques payable to Quicksilva Ltd.
Send to Software Studio, Quicksilva Mail Order, P.O. Box 6, Wimborne, Dorset BH21 7PY. Telephone: (0202) 891744

VISA Access

Boots, W.H. Smiths, J. Menzies, Microdealer, HMV, Hamleys, John Lewis Computers for All and all reputable specialist computer stockists.

SPECIAL OFFER

GSH

MicroSpeech, stated that the unit was compatible with the printer and so "there's no worry about being unable to list out hard copies of programs while the MicroSpeech is in use".

Could you please explain how the printer and the MicroSpeech can be used together, because the Currah unit requires a full-size edge connector, and does not have its own edge connector (to facilitate further expansion), and the printer has a ZX81-size connector?

David Wooding, Cardiff

Okay, we blew it. More Humble Pie, anyone? Ed.



MISTAKE!?!

Did anyone else notice the mistake on the cover of *Your Spectrum* issue 2? The QL photo had the 'L' and 'W' keys swapped.

J Jack, Rochester

Don't let our photography department hear you say it was a mistake — it took them ages to mess about with the keyboard! Ed.

CLOCK THAT

Further to the *Clock This* program (issue 2), I have re-written the alarm routine as, after much thought, I consider the whole screen turning red to be unsatisfactory. Instead, the border still turns red and a two-toned BEEP sounds for 20 seconds — a bit more like an alarm. (The new listing is shown alongside.)

Ian Turtle, Loughborough

		ASSEMBLER	COMMENT
10	CLOCK	RST 56	
20		DI	
30		PUSH AF	
35		PUSH BC	
40		LD A, (65535)	
340	ALARM	LD A,2	Change BORDER colour
350		CALL #229B	
360		LD B, 10	Number of BEEPs in alarm
370	LI	PUSH BC	Save B from overwriting
380		LD A, 1	Load A with length of note
390		CALL #2D28	Put A on calculator stack
400		LD A, 20	Load A with pitch and place
410		CALL #2D28	on calculator stick
420		CALL #03F8	Call BEEP routine
430		LD A, 1	Second BEEP note
490		CALL #2D28	
450		LD A, 25	
460		CALL # 2D28	
470		CALL #03F8	
480		POP BC	Recover B and complete note
490		DJNZ LI	
500		LD A, 62	As before, except extra
510		LD I, A	POP BC
520		IM 1	
530	END	POP BC	
540		POP AF	
550		EI	
560		RET	

```
  10 CLS : CLEAR 65119: RESTORE 300
  15 PRINT TAB 4;"Alarm Clock-I.D.Turtle";TAB 5; FLASH 1;"Loading machine code"
  20 LET sum=0: LET i=1
  30 READ a: IF a>255 THEN GO TO 50
  40 LET sum=sum+a: POKE 65119+i,a: LET i=i+1: GO TO 30
  50 IF sum<>a THEN GO TO 200
  60 LET sum=0: LET i=1
  70 READ a: IF a>255 THEN GO TO 090
  80 LET sum=sum+a: POKE 65128+i,a: LET i=i+1: GO TO 70
  90 IF sum<>a THEN GO TO 200
 100 INPUT "Hours?";h: POKE 65532,h
 110 INPUT "Mins?";m: POKE 65533,m
 130 POKE 65534,0
 140 POKE 65535,0
 150 PRINT ''"Don't forget to"'' INVERSE 1;"RANDOMIZE USR 65120"
 160 PAUSE 200
 170 NEW
 200 PRINT "Error": LIST 9999: STOP
 300 DATA 62,9,237,71,237,94,201,911
 310 DATA 255,243,245,197,58,255,255,254,0,40,6,61,50,255,255,24,97,62,49,50,255
 320 DATA 255,58,254,255,254,0,40,6,61,50,254,255,24,79,62,59,50,254,255,58
 330 DATA 253,255,254,0,40,6,61,50,253,255,24,61,62,59,50,253,255,58,252,255
 340 DATA 254,0,40,6,61,50,252,255,24,43,62,2,205,155,34
 350 DATA 6,10,197,62,1,205,40,45,62,20,205,40,45,205,248,3,62,1,205,40,45,62,25,205,40,45,205,248,3,193,16,226
 360 DATA 62,62,237,71,237,86,193,241,251,201,14404
```

Ian Turtle's re-worked listing for Clock This — the assembler listing is given at the top of the page for your guidance.

DO THE CONTINENTAL

I've read your first issue and found a great many interesting features, for example, *Debugit*, *Go Faster*, *Moving Graphics* and *Sir Clive* (if only for the excellent picture). I also very much liked all the machine code you were able to include.

I did notice, though, that you were missing a letters page and an exhibition list. Also, despite your lack of a club page I would like to tell you a little of our computer club. The Belgium User Group for Sinclair computers is a club which covers the Flemish speaking part of Belgium, and holds regular meetings in Antwerp, Brussels, Ghent, Kortrijk, Leuven and Hasselt.

For more information on the club's operation, you can call me on 03 449 4445 after 7pm (my time please) or else write to the address given below.

P Ob de Beeck, Chairman
BUGS, Drabstraat 144,
2510 Mortsel, Belgium

Question: how can you publish reader's letters in a first issue? Answers on a postcard please, etc, etc. Ed.

GOING WITH THE FLOW

Your magazine fell into my hands quite by chance — it was the only copy left in the shop, surrounded by the usual selection of 'glossy' computer mags. Although already well-thumbed by the rising generation of computer buffs, I was only too pleased to part with my 95p.

Your Spectrum is the type of magazine I've been looking for since I purchased my Spectrum 48K last February. I trust you will keep up the very high standards you have set yourselves.

Thank you for the listings, which are clear, readable and well laid-out. Only one thing missing here (which no other magazine does anyway) and that is suitable flowcharts to accompany the programs.

I know all you experts can do these programs standing on your head while sipping a pint and cracking walnuts between your fingers, but I'm sure there are many of us who would like more information — so that we can follow the reasoning of the program and perhaps even make our own modifications. Flowcharts would certainly help in this respect.

W Jones, Kent

It's an interesting idea. First thing though, we've got to make sure all our progams actually 'flow'. Ed. YS

RICHARD SHEPHERD SOFTWARE

A HOME BUDGETING & BANKING SYSTEM THAT REALLY MAKES YOUR 48K SPECTRUM WORK!



A professional style multi function Cash Controller program that can load and make an entry in just 90 SECONDS . . . thanks to its ZX MICRODRIVE COMPATIBILITY. If you're not ready to upgrade to Microdrive yet, this easy to use cassette program can handle up to 400 transactions with your Spectrum, almost filling the 48K memory of your computer.



BANK ACCOUNT

- Statements on demand from any date to date.
- Holds up to 400 transactions.
- Standing order facility.
- All transactions can be automatically coded to allocate up to 16 budget headings.
- Search facility for any single item by description or amount.

HOME BUDGETING

- 16 budget headings, e.g. Gas, Rates, Car, Tax - you can choose the headings
- Highlights under/over spending
- Full budget expenditure breakdown
- Budget "re-think" facility

LOAN/MORTGAGE CALCULATOR

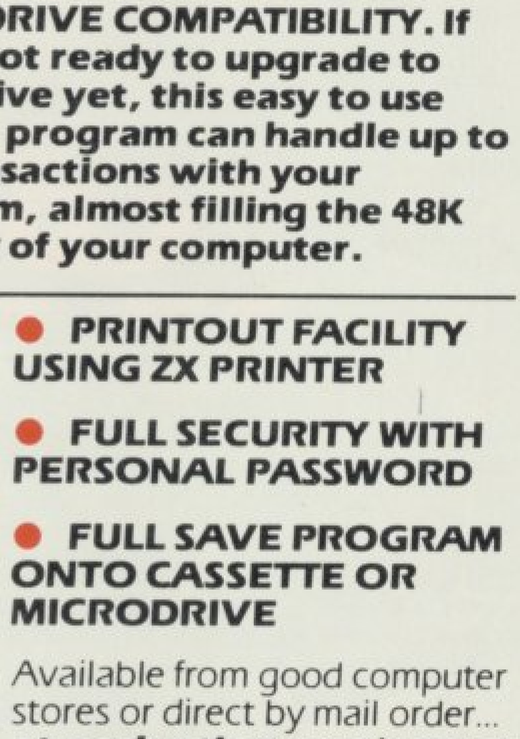
All you ever wanted to know but didn't like to ask! Calculates:

- **Interest rates**
- **Payback periods**
- **Capital sum**
- **Period of loan**

REMEMBER THIS CASSETTE CAN ALSO BE USED WITH ZX MICRODRIVE

- **PRINTOUT FACILITY USING ZX PRINTER**
- **FULL SECURITY WITH PERSONAL PASSWORD**
- **FULL SAVE PROGRAM ONTO CASSETTE OR MICRODRIVE**

Available from good computer stores or direct by mail order... **at a price that won't upset the Bank Manager!**



Credit Card Hotline (06286) 63531

FIRST FOR MICRODRIVE

Dealer enquiries welcome. Generous discounts.

PRIORITY MAIL ORDER

Please send me ________ CASH CONTROLLERS at £9.95 each including post & packing (overseas orders add £1.50).

Name ____________________

Address ____________________

____

____

I enclose cash/cheque/P.O. payable to Richard Shepherd Software

Or credit my Access/VISA Card Number

Signature ____________________ YS3

MICRODRIVE CARTRIDGE NOT INCLUDED

RICHARD SHEPHERD SOFTWARE

ELM HOUSE, 23-25 ELMSHOTT LANE, CIPPENHAM, SLOUGH, BERKS. TEL. (06286) 63531

MISFIRES OF THE YEAR

1983 has seen a large growth in both software games product and the houses that supply it. And with this growth has come an interesting change in advertising techniques — for now, it seems standard to start promoting games at least a month before they become available. Fortunately, for those whose mouths cannot stop watering, much of the software released in '83 was of a high quality. Some, however, fell rather short of the mark and *Gavin Monk* adds his pennyworth to the moans.

Imagine produced the first disappointment of the year with the release of *Schizoids*. After the success of *Arcadia*, everybody was expecting a stunning new game and *Schizoids* was said to offer "A unique experience in arcade action, real-time animated 3D graphics for a breathtaking visual experience and all those quality features you've come to expect from Imagine". However, what we weren't told in the blurb was that the game was only in black and white! Perhaps Imagine flogged its colour television to pay for all the massive four page advertising that went on.

The game idea itself seemed strong enough. You're stranded in space and have the job of interstellar refuse disposal consultant (binman). That means you patrol around a black hole in your space-dozer pushing rubbish into it.

Other than the lack of colour, first impressions with the graphic quality of the product were good. The debris moves around the screen very smoothly, all in animated 3D. But the trouble starts as you realise that your space-dozer is impossible to control (perhaps explaining why Imagine included an order form for a Fuller joystick with the game). And strangely enough, you don't score any points for pushing the garbage into the hole -in fact, there's no real way of scoring points at all. Your tally is simply equivalent to the time you manage to survive. I got my highest score by pressing the start button and popping off for a cup of coffee!

Richard Shepherd Software released *Transylvanian Tower* at the beginning of 1983 — a move that was quickly followed by rave reviews in all the computing magazines. Now, a year on, *Transylvanian Tower* is still being advertised as "A spine chilling adventure". That's as may be, but the plain truth is that the game just can't compete with the new machine code adventures which are now in production. You'll have noticed that I said 'machine code', because although *Transylvanian Tower* loads as code, it's written completely in Basic and not too well at that. No wonder the program is so slow and the graphics move at the speed of attacking snails.



"I hope the real thing's more exciting".

Virgin Games appeared at the Earl's Court Computer Fair in the summer to herald the launch of its first Spectrum games. These were *Yomp, Sheepwalk, Golf* and *Starfire*. Written mainly in Basic, they sold at the unusually high price of £7.95 each. The games proved very slow and, adding insult to injury, turned out to contain bugs. On several occasions I was faced with error messages like 'Integer out of range' and 'Out of screen'.

Yomp is just another version of *Frogger* but with a change of story. Now the idea is to guide your soldier first across a road of speeding Army trucks, then across a minefield. This turned out to be the best of the four and even contained some machine code routines. As for the others, well, in *Sheepwalk* the game characters are depicted by very simple graphics. In fact 'walk' is the operative word because, judging by its speed, the 'dog' obviously suffers some functional difficulty with its legs. *Golf* might have been a good game, except that whenever the ball gets hit off-screen, an error signal pops up and it's time to re-load. I can't say anything about *Starfire* because my copy positively refuses to load at all.

Ocean's *Kong* got to Number One just a few months ago — possibly as much a reflection on its heritage as anything else. Actually, the general program is fine . . . it's the finishing that's sloppy. When Kong jumps up and down at the start of the game, he manages to leave two pixels of leg behind him. Also, your hero seems to suffer from a disability which prevents him from climbing ladders properly! And the message '1 LIVES LEFT' leaves a lot to the imagination. Perhaps the program was rushed on to the market for some reason and failed to receive the customary polish. At least this hasn't happened to the company's latest release — *Hunchback*.

Microl decided to launch *Valhalla* under the trading name of Legend — presumably because its alter-reputation lies in the business software arena. It was advertised as "The program you have been waiting for, the successor to *The Hobbit*". After initial delays, copies of *Valhalla* began slowly filtering through and magazines published reviews that generally rated the program a great success.

Certainly, the operating system is of excellent quality and a pleasure to use and the game idea also seems a good one. The trouble is that it lacks the depth and problem-solving aspects of a true adventure game. The graphic pictures for each location, although excellent, are almost identical. The characters, well animated though they are, get greatly confused when they pass each other. And objects like wine and food just appear as dots on the screen and are very hard to see. Perhaps the fact that Legend described this program as the "successor to *The Hobbit*" raised expectations too high.

So, why were all the reviews so good? Well, with all big launches like this, every magazine wants to have the first review published and the trouble is that checkouts on adventures just cannot be rushed. Reviewers were pushed to write their reviews in time and hence never really got their hands dirty. A certain reviewer for a well-known weekly computing magazine confided that "I only just had enough time to load *Valhalla* — let alone review it".

All griping aside, much of the software released last year showed an extremely high standard with companies like Ultimate Play The Game leading the field. Further progress is assured in 1984. VS

THE YS STRANGLED TURKEY AWARDS 1983



In response to an almost infinitesimal demand from its readers, *YS* is proud to present (albeit rather late!) its games software Strangled Turkey Awards for 1983.

Yes, last year saw it all — the strokes of sheer keyboarding genius, the awful averageness of the Atari copyright lookalikes and, of course, the few instances of near-breathtaking banality.

As you may have gathered, it's the final category that interests us here, and staff on *YS* have deliberated long and hard over the winners of its much-coveted bottom five awards. Now, at long last, the truth can be told! Here, in ascending order of awfulness, are our five top goofers of 1983 . . .

1
Yomp/Virgin Games

A slow, mostly Basic game that became the butt of many a joke in 1983. It was just so naff, it had to grab number one position.

2
Transylvanian Tower/Richard Shepherd Software

Completely written in Basic, and not too well at that. You'll get more 'spine-chilling' adventure out of a melting ice-cube.

3
ET/Hilderbay

Another Basic game with a slice or two of token code. If this had arrived for the Play Power *section of* YS, *the tape would have been sent back!*

4
Schizoids/Imagine

The biggest disappointment of 1983. Although it's written in "100 percent super-fast machine code", its thrills spilled out somewhere along the line.

5
Maze Panic/Silicon Software

It's rumoured that more copies of this game were sent out for review than were sold. Yet again, mostly in Basic with just a twitch of machine code. Silicon Software went bust half way through last year . . . thankfully!

GET THE RIGHT ANGLE ON YOUR SPECTRUM

Trickstick turns your Spectrum into the most sophisticated games machine in the world. Your fingers rest on the Trickstick's six sensitive pads (four directions and two fire buttons) and the harder you press the faster you go or the harder you turn.

Trickstick works by picking up mains hum from your body and converting it by an ingenious circuit design directly into digital input. Proportionality gives vast possibilities for more interesting games.



- Each Trickstick comes with its own interface included in the price.
- Up to eight interfaces can be stacked into each other for 1-8 simultaneous players. (Own 1-8 player game is Attaktics at £10).
- Works with most Kempston compatible software, and with our new programmable adaptor (£10 for Trickstick owners) it works with ANY software.
- Easy to program, even for proportional games.
- In the shops soon. Now available by mail telephone order at £34.50

TRICKSTICK TRICKSTICK TRICKSTICK TRICKSTICK

TRICKSTICK- The revolution that Runs Rings round ordinary joysticks

DOUBLE STANDARDS ?

Although somewhat diverted by the oft-misleading use of the word 'standard', especially when applied to RS232 interfaces, Ian Beardsmore _presents two programming tips from_ YS _readers._

In olden days, standards were things that could be trusted and relied upon. Now they tarnish and crumble in our brave new world. But surely, I thought, computer standards still stood erect and intact? Why, only the other day, I'd been ruminating on the RS232 and all the exciting peripherals you could hang on it.

I decided to carry on thinking but now, several days older, sadder and wiser, all I can do is sit and reflect on the time spent trying to sort out the chaos. Let me quote from the provisional manual of the QL:

> "Unfortunately, the RS232 'standard' shows itself in a large number of different forms on different equipment, and it can be a tedious job, even for an expert, to connect together for the first time two pieces of supposedly standard RS232 equipment . . ."

Now they tell me! If you're thinking of rushing out to buy an RS232 for your Spectrum, then just a word or two in your shell-likes. Be careful, especially if you're just looking for a general purpose interface, and have no specific task in mind. However versatile and useful it *may* seem, the plain fact is that connecting your RS232 interface to the RS232 interface on a peripheral may be as difficult as trying to plug a three-pinner into a light socket! On the other hand, maybe you get a kick out of re-wiring connectors. Worst, of course, is the damage you can cause to equipment. So, care is clearly necessary, but what specifically is it that we have to be so careful about . . . what are the problems?

TOO WIDE A SCOPE?

First, the theory. The RS232 is a 25-line standard based on the 25-line D-socket or EIA. Unfortunately, only three of the 25 lines were defined. These are RX Receive Data, TX Transmit Data and GND (a common ground line). The real problems start just about here. Though 25 lines were made available, with only three specified, the scope for 'flexibility' is enormous. Computer manufacturers, with Uncle Sir Clive not the least among them, have taken far greater liberties and given far less rope. With 22 free lines to play around with, chaos can and has ensued.

Naturally, the first thing to go was the big (and expensive) EIA. Why pay for something that size when you're only going to use five lines? That's why the RS232 can now be seen masquerading as a 12-line edge connector, various types of DIN socket and (the type most likely to be known to Spectrum owners) the 9-pin 'Joystick'-type port. It's quite possible to get two RS232s with the same connector, but wired with different lines — or equally the same lines wired to different connectors.

There are two aspects to the RS232, each of which causes problems. The first lies with voltage, and the second with the lines carried (or perhaps one should say 'the optional lines implemented'). The problems of voltage are potentially the more damaging so we'll take a look at them first.

Paradoxically, the RS232c comes the nearest to being the true standard. It can carry plus and minus 12v and thus is usable with a standard modem. However, like everything else we've been looking at, even this is not exactly standard; most modems can work on 9v or even less.

Next, something that can really cause problems. Another variety of the RS232 is the interface that can only carry 5v and 0v; here its use is therefore limited to TTL applications. Of course, that's fine if you have a special TTL-type interface on your modem or printer. On the other hand, it doesn't take too much to imagine what happens if you get this little detail wrong. It's not a pretty sight to see 12v or even 9v passing through a low-power TTL board; it adds a new meaning to frying chips, and if the positive voltage don't get 'em, then rest assured that the merest hint of a negative voltage will!

More recently a compromise has appeared — the RS423 which is rated at plus or minus 5v; indeed Acorn has been busy telling us how good it is, and up to a point the sentiments are right. But do we really need another standard of a standard? And if you are going to have +5v, why not have the +12v that's more commonly used? (Answer: because the computer power supply is not up to it — therefore, the solution is to invent a new standard that fits the restrictions.)

ARE YOU RECEIVING?

So that's got the voltage out of the way — now let's see what signals need carrying. As usual, a certain amount depends on the use you want

A PEEK IN TIME . . .

A letter from Mr L E Mada reveals a much neater way of PEEKing the two-byte system variables. Instead of using the traditional form shown in the manual, ie. PEEK n+1 *256*PEEK (n+1), he suggests you try the following — namely, PEEK n+1*256+PEEK n. It's slightly easier to type in and takes up two less bytes of precious memory. Mr Mada goes on to say that his way of doing things can really speed up a routine's operation.

In the accompanying listing, there are two routines — the first utilising Mr Mada's ideas and the second following Sinclair Research's suggestions. Using wrist-watch timing, Mr Mada reckons the first routine runs about a second faster. (Type RUN to access the first routine and RUN 16 for the second.)

```
5 FOR a=1 TO 1000
10 PRINT AT 10,10- PEEK
   23731*256+PEEK 23730;" ";a
14 NEXT a
15 STOP
16 FOR a=1 TO 1000
20 PRINT AT 10,10; PEEK
   23730+256*PEEK(23731);" ";a
24 NEXT a
```

In fact, I make it about six-tenths of a second faster, but — never mind — this is just the sort of feedback I'm after. Unfortunately, in this particular case the effect is slightly muted because it's quite possible to use the Sinclair Research form without the brackets, thus removing the main advantage (that is, being less fiddly to key in). Nevertheless, thank you for writing.

to make of your interface; and keep in mind too that the TX Transmit Data signal at one end needs to be linked with the RX Receive Data at the other. Other than the three specified signals the most common (and useful) additional ones are:

RTS : Request To Send
CTS : Clear To Send (on Interface 1)
DTR : Data Terminal Ready (on Interface 1)
CD : Carrier Detect

If the bulk of these signals are available, then usually any missing can be re-routed to a spare existing one, via software.

So where does this leave the Spectrum owner? Well, if you're seriously considering buying an RS232 as a universal panacea, then I know just the right man in a little white coat for you to see! Exaggeration, of course, but I hope that at least you'll have some understanding of the situation — and the problems.

But — a thought — why should anyone go out and buy Joe Bloggs' RS232, when purchasing an Interface 1 complete with RS232 is at last beginning to appear as a credible alternative. It's true! I fail to see one single major advantage of the RS232 over the Centronics. The Interface 1 can send and receive data, and it's a two-way interface . . . except, wait a minute, there *have* been some obscure exceptions that only work in one direction but still call themselves RS232. Uh-huh, you've guessed it, Interface 1 just happens to be one of these obscure exceptions!

THE SOFT SOLUTION

The RS232 on the Interface 1 is really only of use as a printer interface, where data is sent in just one direction. But, have you seen the price of RS232 boards for the common Seikosha and Epson printers? Change printer manufacturer and you have to buy a new board. So why not get a Centronics interface instead, usable on all the common printers and probably cheaper into the bargain!

In theory, there is a glimmer of hope with this problem. The fault lies in the fact that much of the work of the Sinclair RS232 is done by software, and the problems lie with the DTR line. If this line is high then it will transmit data; if it's low then it stops the interface dead, effectively disabling it. At present, I'm looking for ideas of a way round this — in theory, anything in software can be changed, even if the software is in ROM.

Then again, some of you Spectrogenii might want to get clever and try re-wiring the plug and socket so the computer thinks that DTR means an incoming signal. That's not bad because it will allow a signal into the computer when DTR is '1'. You'll not be able to do anything else with it, but never mind. The real fun takes place when at the end of the data, the DTR will go low; this time effectively disabling such insignificant items as keyboard scan and display. On the supposition that a computer with a blank screen and dead keyboard is not quite what the doctor ordered, let's see if there's another way around the problem.

A gentleman to the last, I will allow Sinclair Research to make the final example of the problems created by the confusions of this standard. The RS232cs on the QL appear far superior and, having their own control chip, should allow full duplexing. And, in an effort to make the interfaces compatible with as many others as possible, they are configured in opposite ways. Thus, Ser 1 looks very like I/F 1, and so does Ser 2 except that it's configured backwards. This harks back to what I mentioned earlier about re-wiring the Interface 1's RS232. In this configuration, TX Transmit Data becomes an Input command because transmit now means putting in; while RX Receive Data, of course, means the opposite. Thus, when you are receiving data, *we* are putting it out. Of course, in this situation Receive Data (out) could now be connected to Receive Data proper because the same now means the opposite; and in the same way TX meaning RX can be connected to TX that actually means TX — if you get my meaning! YS

JUST LIKE AUNTIE

Mr K Cooper has come up with something for all you structure freaks. It's a way of simulating the BBC Micro's (*Who he? Ed.*) PROC and ENDPROC commands. Apparently the idea is not exactly new, just hardly ever seen.

All it entails is assigning a variable name to a number — a number that's used as the line number of a GO SUB routine. In this simple demonstration program, the routines are called DISPLAY and column II (or perhaps I should say the PROCedures are called DISPLAY and column II).

Mr Cooper adds that while he does not particularly go in for structures in a big way, it was useful in de-bugging his demo program. In the listing he sent in lines 50 and 60 had been transposed.

Although not wishing this page to become embroiled in an argument for or against structuralism, it is interesting to ponder on the number of Spectrum owners who might find problems if they upgrade to the QL, which probably has more structure forms that the BBC Micro!

```
  10 REM   DEMONSTRATION
  20 REM      (Set Up)
  30 REM
  40 LET a=65
  50 LET x=0
  60 FOR b=1 TO 26
  70 LET DISPLAY =1000
  80 LET columnII=2000
  90 REM
 100 REM   MAIN PROGRAM
 110 REM
 120 IF b<14 THEN  PRINT b;" ";
 130 GO SUB DISPLAY
 140 NEXT b
 150 STOP
1000 REM      DISPLAY
1010 REM     (columnI)
1020 REM
1100 IF b>13 THEN  GO SUB columnII
1110 IF b<14 THEN  PRINT CHR$ (a)
1120 LET a=a+1
1130 RETURN
2000 REM
2010 REM    DISPLAY columnII
2100 PRINT AT x,12;b;" ";CHR$ (a)
2120 LET x=x+1
2130 RETURN
```

If you've any problems with hardware or software on the Spectrum, or you just want the space to air your programming genius, drop a line to Ian Beardsmore, Your Spectrum, 14 Rathbone Place, London W1P 1DE. *Any feedback would be welcome — come on, let's hear from you!*



. . .this is Jamie's first floppy, that's Alison's first printout and here's Alex in his first User Group. . .

YS HARDWARE REVIEW:

THE TOWER OF POWER



IAN McKINNELL

Towering above competitive add-on units for the Spectrum, the Basicare system offers a wealth of new applications for the 'professional' user. *SQ Factor* finds out whether the system measures up . . .

The early adverts for the mould-breaking ZX81 suggested that the Cambridge miniscule miracle was capable of controlling a power station. While others choked, turned purple, or rolled chortling in the aisles between the card sorters of their doomed mainframes, Basicare quietly designed a powerhouse that the ZX81 could run — a modular one, limited only by the height of your ceilings. And now it's available for the Spectrum.

The ceiling height is significant because the expansion modules stack neatly on top of each other, and because the accompanying literature gives several different figures for the maximum memory that can be added to your ZX, ranging from 256K to a gigabyte (this last from a reprint of a review by another magazine and hence, highly suspect). At just under an inch and 64K per module, a one megabyte memory will stand about a foot and a half high, counting the two modules needed for interface and paging. Personally, I'd stop there, but a real maniac with six feet of clearance between desk and ceiling, or who is willing to compute in a stairwell will be pleased to know that the Basicare documentation definitely says that four megabytes of address space is possible. Not all of this is available for RAM, however, as the paging system devotes some areas to other uses.

Non-maniacs who have not yet left us may well be more interested in these other options: such as the eight-channel A/D converter; the non-volatile, auto-start CMOS memory module; Centronics interface; EPROM module; and more. There are even hints of a 16-bit add-on processor to come, but at that point you'd probably sling out the original computer as a drag on the system, and run the expansion on its own!

READ WHAT'S WRIT

Us aged duffers, hands crippled with solder burns and minds cluttered with the characteristics of thermionic valves, nevertheless possess one clear advantage over most whizzkids when it comes to coping with new toys. Endless and repeated experience has taught us to read the instructions before doing anything else. Basicare's literature is nostalgia reborn. I love it. Phrases like "Connections to the aluminium strips can be made with small crocodile clips" brings small crocodile tears to my eyes. Before computers became consumer products we enthusiasts drooled for hours over documents like these in search of enlightenment. It does eventually come, provided you are sufficiently enthusiastic. The information is all there, but it's not too well presented.

What you get is a flock of photocopied 'technical sheets', the mix depending on which modules you have acquired. Most of them were originally written from the ZX81 and are accompanied by an update sheet for the Spectrum, to be read in conjunction. The business of memory paging, without which no eight-bit micro can address more than 64K, is not beyond the understanding of anyone — but it's beyond the patience of lots of people, I reckon, especially if they have to work from this documentation. Some of the confusion arises from uncertainty about whether the piece of paper you're reading is meant to apply to the Spectrum or the ZX81 or both, particularly where addresses are concerned. These and other numbers are also given sometimes in Hex and sometimes in decimal, with no indication of which is which. However, most of the complexity is in the concept itself and, therefore, these pro-

ducts should be regarded as not for beginners, unless the beginner is determined to become an expert the hard way.

GOING ORGANIC

The 'Organic Bus' is a marvel of tidiness compared to the breadboard tangle that this sort of equipment usually results in. Stacking the modules chains them onto the bus via a 64-way plug/socket out of the top and bottom of each module. Once plugged together, they stay put very firmly and the stack can be handled as a unit.

All the modules have pins sticking out the back which are used for various purposes. On the memory modules, pins are linked together to select appropriate positions in the memory maps, while the in/out modules also use them for guess what? Linking clips are supplied, placed in 'get-you-going' positions. The Centronics port has a proper 26-pin connector with grips.

The edge connector that plugs into the Spectrum has something of a home-made look, being held together mostly by a large dollop of cold-pour rubber

THE BASICARE SYSTEM DISASSE

FOUNDATION MODULE

Persona SP

This is the main module whereupon all others are stacked — the foundation of the tower block. By itself, it does nothing. Its function is to provide a buffered 64-way bus, regulated power supply (using the unregulated power from the Spectrum's supply, via the edge connector), and various decoding and paging signals for the other modules.

The buffering is a good idea from a safety point of view. I accidentally plugged the expansion into my 48K Spectrum without having previously made an essential modification to the innards (of the Spectrum) and got away with having two different lots of memory chips trying to control the data bus at the same time. Not recommended, but not a disaster either.

The 16K Spectrum mates up to the Basicare system with no hardware mods but the 48K version has to have a connection made internally between one pin of a decoder chip and the edge connector. The purpose of this is to turn off the top 32K of memory whenever the Persona is attached. Unfortunately, this part of the memory can not be used with the Basicare system; it does seem a little odd to start an expansion by removing 32K of RAM and then presumably replacing it externally, and this feature will make the system rather less attractive to users of the larger model.

The Persona can power up to five modules, using the power pinched from the Spectrum supply. Larger expansions will require an additional power pack to be attached to the Persona.

CONTROL MODULE

Pericon A

Control applications requiring a lot of TTL in/out can be implemented with this module, or for heavier use, the buffered version, Pericon B. There are three eight-bit ports which can be either input or output at your option, giving 24 in/out lines. If you want any more, you just slap on another module, up to a limit of perhaps four. Well, at least four, say. It would seem that more might possibly be managed using paging, which requires a Minimap module.

The technical sheets on this one are technical indeed. Suffice to say that the inputs and outputs are low power Schottky TTL equivalent (can't be bad), and



compound; but it is serviceable.

Further Spectrum add-ons are not catered for, as there's no carry-through of the edge connector. However, Basicare says the system is completely compatible with the Microdrive, and it did indeed work with ours. The two would seem to be good companions, with the Network and RS232 of the Microdrive filling a gap in Basicare's range.

I asked whether the Prism VTX5000 Prestel adapter was compatible but Basicare has yet to investigate the idea, and I am too chicken to just plug it all together and see if it melts. For the moment I access Prestel with an unadorned Spectrum plus VTX5000.

Two further modules which we did not review are Sonus, a three-channel sound generator, and Toolkit, which can house 8K of your own utilities in EPROM. Other developments are promised for the future.

OVERALL

What we have here is a failure to communicate. The Basicare gear is tidy, powerful, robust and complex. It can give Sinclair users access to advance

SE MBLED BEFORE YOUR VERY EYES

that up to 2mA can be had on output for driving a transistor, LED, or whatever if you don't mind the voltage dropping below TTL levels. A few circuit examples are included to get you started, driving relays or the like. There is even a fairly ambitious project for making yourself a membrane keyboard out of kitchen foil and Sellotape (plus small crocodile clips). This is the kind of experiment that I prefer to conduct mentally, a la Einstein, but it would not be out of place in a classroom.

From the computer end, the ports appear as simple memory locations and are accessed by PEEKs and POKEs. Complexity appears in the shape of the control port which is used to set the others to input, output or modes beyond, strobing and even bi-directional modes being available with a reduction in the number of lines. At this point I moved on to . . .

CENTRONICS MODULE

Pericon C

This is the Centronics interface module for use primarily with proper size printers. I was unable to test it as I have no Centronics equipment about the place (we're RS232, we are). However, I'm forced to recommend it anyway, as you'll be unable to use the cheap and cheerful Sinclair printer with any of this gear. Nothing exotic like the clash of mismatched memory maps, merely the fact that the Persona module does not have a carry-through edge connector for the attachment of further Sinclair stuff; also, the one on the Sinclair printer, which does allow chaining, is not the full width, so the Persona can't plug into it. Right daft, but who to blame? Both companies seem to have goofed here. You won't be able to fit your joystick, either.

You get a cassette with the Pericon C which when loaded allows you to specify width (maximum 80 columns) and lines per page (maximum 66 lines) and then sets up a machine code driver to run the Centronics printer directly from LLIST and LPRINT. This driver is placed at address 32300 decimal, and I have no idea if it can be moved as the cassette seems to supplant any technical sheet concerned with this module, and there are no REMs in the program. I would have thought this was an ideal candidate for inclusion in the non-volatile memory of the Autostart Drom. As it stands it would seem to limit one to a 16K Basic program, however much more memory you might have. (See also the RAM section on this point.)

Whatever niggles a conscientious (or contentious) reviewer might feel the need to grub up, real black-on-white listings instead of the standard 'silver streak' are a heartfelt need long denied to magazine editors and the like, and any piece of kit that can bring about this Jerusalem cannot be decried.

ANALOG MODULE

Link A

Eight channels of analog input can be handled by this module, and you can, again, have up to four modules at least. The analog voltages must be between zero and plus five volts, and while a simple resistive divider will bring larger voltages into the module's range, other techniques are recommended for accuracy and safety. In particular, the inputs must never be allowed to go negative or over five volts. This is not a problem with resistive probes such as thermistors or strain gauges, but if you're thinking of spectro-analysing your Hi-Fi then it's a point to watch.

Unfortunately, what seems to be a diagram of a ratiometric input arrangement in the 'tech' sheet only had the labels printed, with the drawing itself left out. I would expect that anyone having a use for this module would probably already be familiar with instrumentation electronics, buffer amps and the like, so maybe it doesn't matter.

The computer reads the module with PEEKs and selects the required input with POKEs. Accuracy is stated to be one bit, working out to 0.4 per cent, which is better than any probe is likely to give. This one would seem to be suited to genuine scientific monitoring purposes.

FUNCTIONS MODULE

Autostart DROM

This is the point where the Basicare system came alive for me. Not only does this module give you up to 8K of non-volatile memory for your own routines, but it can actually cut out the Spectrum ROM at power-up, run its own initialisation routine, and leave you with a bunch of extra interrupt driven functions. It came supplied with resident software providing an auto-number routine for Basic lines, a machine code reset (derived from Toni Baker's article in *YS* issue one), and a routine to read the Clock module in a convenient way. I've added my own bit of code to give an instant reference 'free memory' function (see Listing 1) just to show that anyone who wants to re-write Sinclair's operating system can do it with this module.

However, I must confess (having gone on a bit about us old hands always reading the instructions) that I switched off the write protection before I knew what I was doing, and somehow wiped the supplied program. Basicare kindly sent me a cassette with which to restore the code, if not my applomb. As to why it wiped, either I was reading the wrong piece of paper (there were three lots for this module), or I was attempting to alter code which was actually running at the time, once every interrupt; on the other hand, just possibly my 'reviewer mode' simulation of the new user, a necessary ability for an article like this, is more advanced (read 'naive') than I thought.

The module is supplied with internal sockets for four CMOS 2K RAM chips, but you needn't pay for more memory than you need. Ours had 4K fitted and two empty sockets. Write protection is selected for each chip by its own diddy switch at the back of the module, and the autostart feature can be enabled or disabled by the positioning of a link nearby. The write enables should only ever be switched on when you are entering or LOADing in code, and I would personally never touch them without SAVEing the relevant section of memory to tape first.

The obvious appeal of having your own idiosyncratic printer routines, utilities, or alternative character sets all ready on-board and operative at power-up needs no further elaboration, and makes this the most interesting module in the system.

The only snag I found was that the autostart routine that reads the Clock module places the time and date info into the Basic variable, t$. This means that a simple PRINT t$ command will put the time and date on the screen. However, it also means that a RUN or CLEAR command, which clears the variables area, confuses the interrupt routine and drops you into a sort of semi-crash. Furthermore, LOADing in a Basic program also seems to mess up, with the program apparently unable to recognise its own variables. But you can write a program while the feature is active, using t$ to PRINT the time if you want, without any trouble, so I should think the debugging of this feature did not progress much past the EDIT mode. Too bad, because it will stop you using the autostart software unless Basicare fix it.

However, I can exclusively reveal that if you POKE 57866,24, then the t$ update is bypassed and you don't get any more trouble. You don't get the time, either (there are other ways to read the clock),

but you can still use the other added features, auto-number, reset, and my own mem bit. This POKE replaces a conditional relative jump (20 Hex or 32 decimal) with an unconditional one (18 Hex), so you know how to put it back. Don't forget the write enable switches. This dodge is not needed unless the clock module is in place, as all it does is tell the software that there is no clock to read.

CLOCK MODULE

Clock

This is a proper real time clock with battery back-up and its own timing circuitry. It's therefore much more accurate than the Spectrum's internal 50-cycle counter, and it doesn't lose time during BEEP, etc. Besides the time, down to tenths of a second, it also keeps track of the day, day of the week, month and year, with the appropriate software. A cassette is supplied with two programs, one to set the clock and one to read it.

The internal nicad battery will keep the clock going for three months on a full charge, and it charges automatically.

Our module didn't work at first, in fact the whole system hung, but it only needed setting. We couldn't load the setting program because it hung on power-up, as we were using the autostart module, which accesses the clock every 50th of a second — said clock being hung up because it needed setting. Get the picture? No problem at all in fact, and a line of text in the documentation would have saved us having to phone Basicare (again) and do our increasingly familiar moron act. If it happens to you, just ditch the autostart long enough to use the setting program. The other problem with the autostart and clock we have already discussed.

MEMORY MODULE

RAM

We got the smaller RAM module, the 16K. A 64K version is available, but requires paging which would require a Minimap module. Unless you are prepared to get involved in such techniques, you're limited to a 16K RAM expansion (not counting the additional 8K of the DROM). This would seem to be a hangover from the ZX81, which had a very different memory map to the Spectrum.

The major limitation here is that paging is only possible with a RAMTOP or 32768 or lower, which is to say within the Spectrum's own on-board RAM. You obviously cannot page the stack in and out of the memory map without giving yourself problems! This plainly means that the mucho memory craziness with which we opened this piece applies only to machine code programs and data that can be stored above RAMTOP. Basic programs longer than the normal 16K versions (which are really restricted to about 9K) cannot be run if you are going to use paging.

However, if you eschew paging for the duration, you can CLEAR to 48983 and still have room for the UDGs. In fact, the ROM does this automatically on power-up. You can then write or LOAD about 15K of Basic. This is short of the standard 48K machine's capability, and you'll not be able to LOAD commercial 48K sofware. To be fair, that is not what the Basicare system is about, but it does seem a little perverse to have a Megabyte of memory, but still not enough room for *The Hobbit*.

The reason a further 16K cannot be simply added on top is that the last 16K of addresses is set aside for the DROM and in/out modules. And that is what Basicare *is* all about. Perhaps there is a way to get a RAM module to sit in that slot. There are, after all, scads of pins sticking out the backs of all these boxes, with which to play all kinds of memory swapping tricks, but I can't find it in the docs, and it's beside the point anyway. For *The Hobbit* you just carefully remove the Basicare equipment and hide it somewhere where it won't get smashed up by adventure loonies.

If you don't quite see what a lot of separate 16K pages that can't be used for Basic are good for, how about storing, say, 24 Hi-res screens and zapping them into the real screen memory in turn to produce a second of Disney quality animation? That would need three 64K modules.

```
0000                 ;MEMCHECK
0000                 ;
0000                 ;ACTION- HOLD ENTER KEY TO SEE INDICATION
0000                 ;OF FREE MEMORY- ONLY APPLIES DURING EDIT
0000                 ;MODE WITH K CURSOR- THIS VERSION MEANT
0000                 ;FOR USE WITH BASICARE AUTOSTART SOFTWARE
0000                 ;BUT SEE BOX BELOW AND BASIC VERSION
0000                 ;
5C3B =               FLAGS   EQU     23611
5CB2 =               RAMTOP  EQU     23730
5C65 =               STKEND  EQU     23653
0000                 ;
E2C5 =                       ORG     58053
E2C5 C326E3                  JP      58150   ;REPLACES RETURN
E2C8                                         ; INSTRUCTION
E2C8     ;
E2C8     ;*******************************************
E2C8     ;* TO ADD MEMCHECK TO TONI BAKERS RESET,   *
E2C8     ;* REPLACE HER 'RETI' INSTRUCTION WITH A   *
E2C8     ;* JUMP RELATIVE TO THE END OF HER ROUTINE,*
E2C8     ;* THEN ADD ON THE FOLLOWING (IGNORING THE *
E2C8     ;* ABOVE AND THE ADDRESSES)                *
E2C8     ;*******************************************
E2C8     ;
E326 =                       ORG     58150
E326 F3                      DI
E327 F5                      PUSH    AF
E328 E5                      PUSH    HL      ;YOU KNOW IT
E329 D5                      PUSH    DE      ; MAKES SENSE
E32A C5                      PUSH    BC
E32B 3A3B5C                  LD      A,(FLAGS)
E32E CB5F                    BIT     3,A     ;K CURSOR?
E330 201F                    JR      NZ,SKIP
E332 3EBF                    LD      A,OBFH
E334 DBFE                    IN      A,(OFEH)
E336 1F                      RRA             ;ENTER PRESSED?
E337 3818                    JR      C,SKIP
E339 ED5B655C                LD      DE,(STKEND)
E33D 2AB25C                  LD      HL,(RAMTOP)
E340 ED52                    SBC     HL,DE   ;CALCULATE MEM
E342 44                      LD      B,H
E343 4D                      LD      C,L
E344 CD2B2D                  CALL    O2D2BH  ;STACKBC    USEFUL
E347 CDE32D                  CALL    O2DE3H  ;PRINTFP  ROM ROUTES
E34A 3EBF            LOOP    LD      A,OBFH
E34C DBFE                    IN      A,(OFEH)
E34E 1F                      RRA             ;ENTER STILL PRESSED?
E34F 30F9                    JR      NC,LOOP
E351 C1              SKIP    POP     BC      ;IT ALL COMES BACK
E352 D1                      POP     DE      ; TO ME NOW
E353 E1                      POP     HL
E354 F1                      POP     AF
E355 FB                      EI
E356 ED4D                    RETI
E358
E358 =                       END
```

```
  10 CLEAR 32330
  20 FOR i=32334 TO 32398: READ a: POKE
i,a: NEXT i
  30 DATA 62,40,237,71,237,94,201,62,63,
237,71,237,86,201,255,243,245,229,213,19
7,58,59
  40 DATA 92,203,95,32,31,62,191,219,254
,31,56,24,237,91,101,92,42,178,92,237,82
,68
  50 DATA 77,205,43,45,205,227,45,62,191
,219,254,31,48,249,193,209,225,241,251,2
37,77
  60 SAVE "mem16k"CODE 32334,66
  70 REM LET on=USR 32334
  80 REM LET off=32341
  90 REM Delete REM from one of the line
s above to activate it
```

A BASIC program giving an interrupt driven MEM function, in answer to Toni Baker's request for a method of using Interrupt 2 on 16K machines. Opaque, isn't it?

techniques for specialised applications which they would be unable to find elsewhere, and all in an integrated system — provided they are able to understand how to use it, that is! What it needs most is a complete re-write of the documentation, with separate versions for ZX81 and Spectrum, preferably written by someone who doesn't do Hex to decimal conversion in their head.

Anyone who merely wants to bump the 16K Speccy up to the full 48K and no more, would be well advised to go for a simpler and cheaper alternative. This route is for the user with an application in mind, and the ability to implement it alone. Commercial programs for the Spectrum that use more than 48K and can cope with paging are not noticeably available.

If you want to turn your Spectrum into a sophisticated tool you'll find the necessary building blocks in this system. The rest is up to you. YS

SEXING YOUR SPECTRUM

An amusing divertissement for any known collection of true Spectrumists is to enquire innocently whether someone can identify a particular ZX model — or ask how A's machine differs from that of B. It's a guaranteed crowd-puller!* Stephen Adams *fancies he can supply the answers and, taking the Spectrum by the chips, he details the various stages of development since its April 1982 launch.

After three issues of the ZX Spectrum and countless modifications, many of you must be wondering just what goes on inside that little box of tricks — and just how you're affected by which model you have. Well, wonder no longer — here for your delectation is a comprehensive guide to the Spectrum's innards.

Perhaps the most obvious distinguishing factor of the Mark 1 is that it sported a grey keyboard. The Mark 2s and 3s both have a blue background to the keys.

Internally, the issues have been mucked about a bit. But, for details check the annotated photos for the complete picture. Apart from the RAM changes and the re-positioning of various major components, Sinclair Research made their most radical alterations to the Spectrum when introducing a redesigned PCB and a new ULA to their final version.

When Sinclair Research began work on the issue 3 Spectrum, a low-powered version of the ULA was employed. Even so, a problem with the Spectrum internal power supply is that one of the transistors (TR4) tends to blow up in response to sudden voltage fluctuations. This results in the loss of the −5 volt and +12 volt supplies to the RAM and the appearance of little black squares over the screen when switched on. The only solution is to change the transistor (of course, if the machine is under guarantee, then send it back to Sinclair Research).

The change in ULA on the Mark 3s also brought to light another problem, because some software houses had not been following the procedures set out in the Spectrum manual. The keyboard input port also reads in a value from the EAR socket and until now this has been Binary 1. On Mark 3 Spectrums, this value was not maintained because, to reduce power consumption, the values of the pull-up resistors were altered.

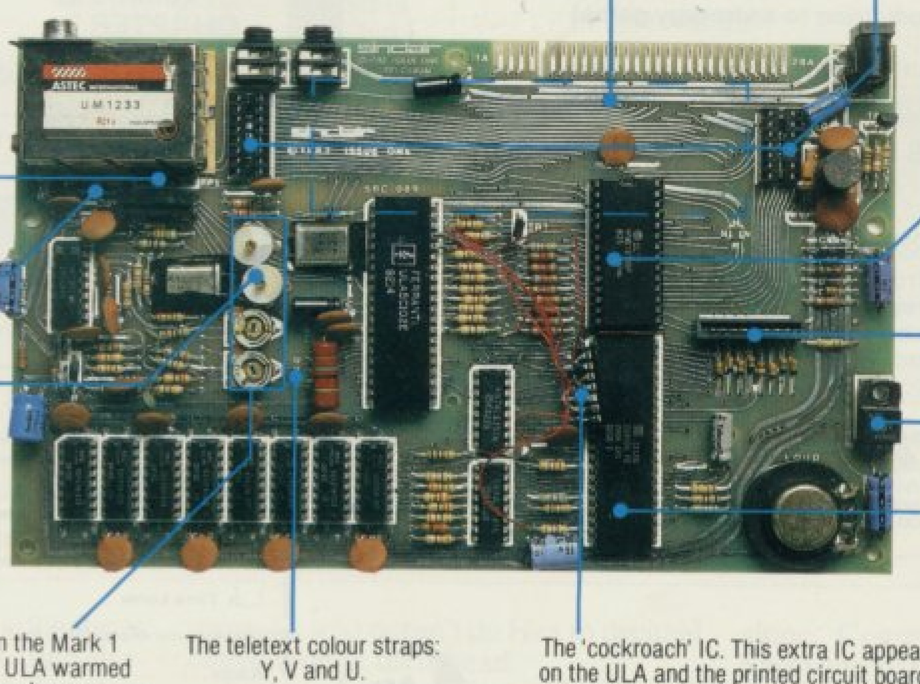
The result is that the EAR bit now floats until the ULA has warmed up, and no value has been given for it by Sinclair Research in the manual. Games and other software which check the whole byte, and not just the keyboard bits, can no longer be used and anyone owning a Mark 3 is strongly advised to check before handing over the 'greenies' that any new purchase is not going to suffer this problem.

It also turned out that Mark 1's design caused problems with the Interface 1 — and thus its compatability with the ZX Microdrive. A transistor modification has had to be made inside the interface to cut down the time taken to switch on the unit from power-up. YS

SPECTRUM ISSUE 1 (16K) — APRIL 1982

On the 48K version, you can easily spot the capacitors on the RAM board that hangs upside down, just inside and above the edge connector. These boards were originally flogged by Sinclair Research as part of a £60 upgrade but because this meant Spectrums had to be returned to the factory, few took up the kind offer and the idea was eventually dropped.

These IC sockets remained empty on the 16K version; for the 48K model, chips were provided to upgrade the on-board memory.



The keyboard data sockets.

The 'VID' strap.

On the Mark 1 Spectrums, to bring the colour back to its former glory, you could adjust the colour crystal frequency — with the help of a screwdriver and via the hole in the underside of the machine.

The device's ROM. Early versions of the Mark 1 Spectrum contained EPROMs, which were very thirsty for power tending to turn them into radiators and cause problems with over-heating. These EPROMs can be recognised by a little 'window' in the top of the chips.

The keyboard address sockets.

The device's regulator.

The Z-80 CPU.

The colour controls. A problem on the Mark 1 and 2 Spectrums was that as the ULA warmed up, it caused the sync pulse to change frequency.

The teletext colour straps: Y, V and U.

The 'cockroach' IC. This extra IC appeared between two pins on the ULA and the printed circuit board, its job being to stop the CPU and the TV interface fighting over the data bus. This alteration was later incorporated into the Mark 2's ULA.

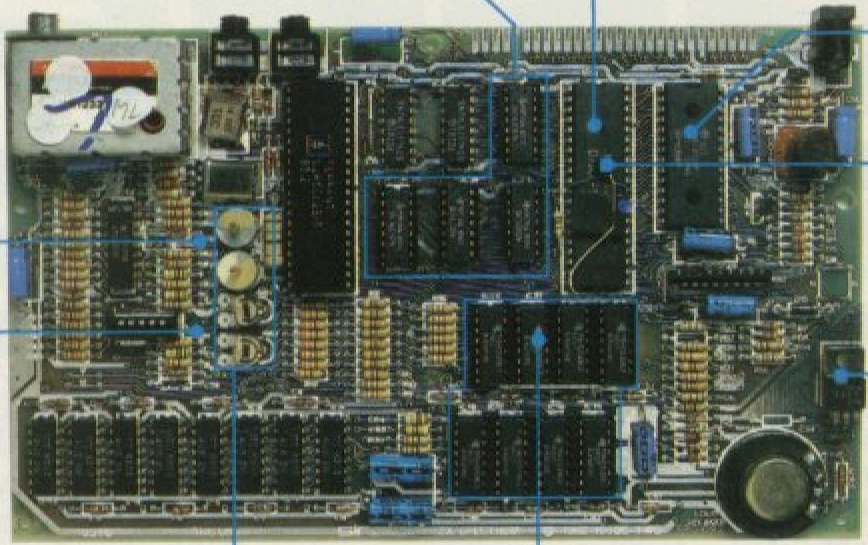
SPECTRUM ISSUE 2 (48K)

AUGUST 1982

The decoding chips. Space is provided for these chips in the 16K version so that other manufacturer's chips can be readily inserted.

The Z-80 CPU. Notice how it has change position from the earlier model.

The device's ROM. This too has changed position from the Mark 1 Spectrum.



The 'VID' strap. The monitor interface which Sinclair Research once boasted of is, in fact, only available on specially modified versions of the Mark 1s and 2s.

The Teletext colour straps: Y, V and U.

A patch was introduced on the Mark 2 to cope with ULA decoding problems.

The device's regulator. Still in the same position as on the Mark 1 board.

The colour controls. On the Mark 2 model, no access was provided from the bottom of the unit — thus tempting frustrated owners to break the guarantee in search of the perfect screen display.

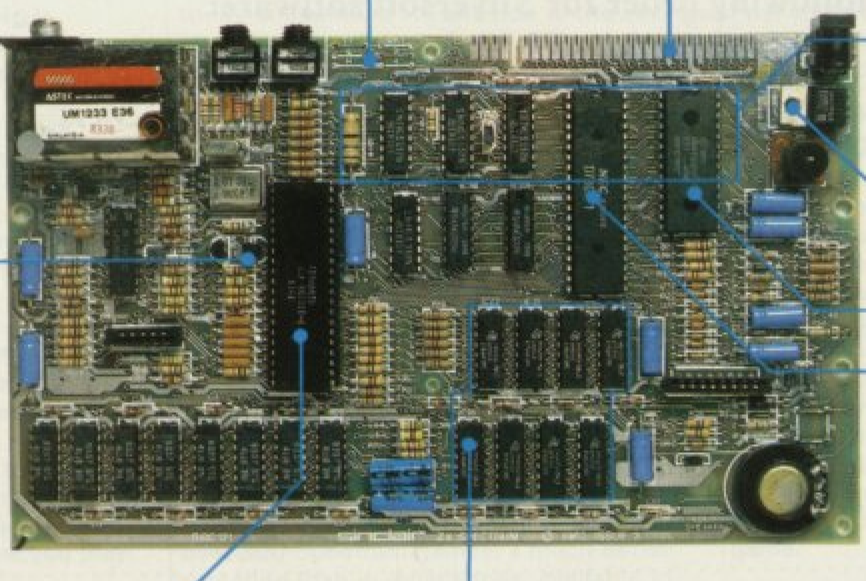
The RAM chips — this area remains unoccupied on the 16K version of the Mark 2, although it did allow users the option of being able to insert the chips. But this modification was again supplied cheaper by Sinclair Research's competitors and the price for the 32K upgrade has now dropped to around £22 for a DIY kit.

SPECTRUM ISSUE 3 (48K)

JULY 1983

The Mark 3 now incorporates straps to allow the manufacturers to select from three different types of RAM; the original Texas chips became unavailable and alternative chips featured different pin layouts. It's possible therefore, when buying a RAM kit, to find that one of the decoding chip's legs is bent or even soldered to a different pin.

To link the Spectrum to a composite signal monitor, run a lead to the VIDEO and 0 volts connections as shown in the manual — preferably via an edge connector. The other connections only give out colour difference signals (B-Y, R-Y and Y) and not the normal RGB; for these, a special monitor is required.



The heat sink has been moved nearer the expansion port and, in fact, can easily be seen through the back of the machine's edge connector port.

The colour adjustment controls and Teletext colour straps were removed on the Mark 3 to be replaced by a facility within the ULA to make the colour self-adjusting. This was important because one of the complaints about the Mark 1s and 2s was that, however much you tweaked around, they simply refused to work with some German and Japanese TVs.

The device's regulator has been moved from its former position on the Mark 1 and 2 versions.

The device's ROM.

The Z-80 CPU.

Both overheating and voltage variations in the mains supply have tended to plague the ZX Spectrum. Some of this was cured in part when Sinclair Research changed to a new low-powered version of the ULA for the Mark 3.

The RAM chips.

Perhaps the most significant change between issues was that the Mark 3's printed circuit board was re-designed by computer. This improvement led to a reduction in 'noise'. But it also drew complaints from various software houses that, when using machine code, they couldn't access the keyboard via I/O instructions as fast as on the Mark 1.

TOKEN OFFER 2

SILVERSOFT/YOUR SPECTRUM

Here we are, at it again, giving somebody else's money away. Not satisfied with only allowing Artic in Issue 1 that 'warm glow of unselfishness', now our tight-fisted ol' tart of an editor has actually persuaded those normally quite level-headed people at Silversoft to part with the valuable stuff.

That's right, for readers of *Your Spectrum* here is the same kind of lunatic offer as before: anyone buying TWO pieces of Silversoft software will get the THIRD piece absolutely free, gratis and zilcho. No joke — that means you lucky people will be saving the princely sum of £5.95!

Here's the titles you can choose from:

Orbitor	**£5.95**
Ground Attack	**£5.95**
Muncher	**£5.95**
Starship Enterprise	**£5.95**
Slippery Sid	**£5.95**
Cyber Rats	**£5.95**
Exterminator	**£5.95**
Brain Dammage	**£5.95**
Armegeddon	**£.595**
Mission Impossible	**£5.95**
Free's Bee	**£5.95**
Robot Riot	**£5.95**
Sam Spade	**£5.95**

HOW TO ORDER

Just fill in carefully the coupon/ Token below, and mail with your cheque/postal order (made out to Silversoft Ltd please) to:

Your Spectrum Token Offer 2
14 Rathbone Place, London
W1P 1DE.

YOUR SPECTRUM — TOKEN OFFER

Please rush me the following order for Silversoft software:

	Title	Price
1.	....	....
2.	....	....
3.	....	FREE

I enclose the cheque/postal order made out to ***Silversoft Limited*** for the value of £________

Please send the software to:

Name ....

Address..

....

....

Postcode..

Date ....



PSI-SPY

EXTENDING BASIC

Stand by to have your mind ever so slightly blown because, for those proud possessors of a ZX Interface 1, Toni Baker is about to explain the procedure by which new commands can be added to the Spectrum vocabulary.

First the bad news — you'll need to do some actual *thinking* if you want to get the full benefit from reading this article. But the good news is that it's worth it because, once you've 'taught' the Spectrum a new statement, then you'll be able to use it in any Basic program you care to write. So, let's move on and look at some fundamental concepts...

The first concept is the fact that the Interface 1 contains a brand new ROM (called the Shadow ROM) which is totally separate from the normal Spectrum ROM. Both ROMs use the same addresses, so you can only ever have one of them in place at once. The Shadow ROM occupies addresses 0000 to 1FFF Hex — so as soon as you start using it, every CALL instruction in this range will actually CALL a routine in the Shadow ROM, *not* the Spectrum ROM. This includes the re-starts — all of the RST instructions do different things when the Shadow ROM is in place. And here's where the crunch comes in: *an Extended Basic handling routine operates with the Shadow ROM in place*.

A machine code routine to cope with a new Basic command has to scan through such a line of Basic to make sure all is well, and then carry out the required task. The second fundamental concept then is that of the 'current character' in a Basic line. The gist of it is that we can only check one character at a time (although in practice, there are some ROM subroutines we can use to speed things up) and so the 'current character' is a pointer to the character we are currently checking. (This pointer is stored as the system variable (CH_ADD).)

To avoid all confusion, we'll precede all Shadow ROM addresses with the capital letter 'X'. So '1234' means address 1234 in the Spectrum ROM, and 'X1234' means address 1234 in the Shadow ROM. Because all the re-starts are different, it'll be useful to spend some time explaining just one of them — RST X10. The byte 'D7' (meaning RST X10) must be followed by two bytes of data. These bytes represent an address in the Spectrum ROM (that's the Spectrum ROM, *not* the Shadow ROM) with, as ever, the bytes in reverse order. The purpose of this instruction is to call a subroutine in the Spectrum ROM at the address given. So 'D72000' — which is Hex for 'RST X10/0020' — actually means CALL 0020 in the Spectrum ROM. This is, of course, very handy.

Let's consider a routine which extends Basic, making 'USR' a command instead of just a function. Thus, the line:

```
100 USR 40000
```

is now a valid Basic line. The purpose of this Basic line is to call a machine code routine (in the above case at address 40000), but it avoids all that messing around with RANDOMIZE USR and PRINT USR and so on. In other words, the final value of the BC register is 'forgotten' at the end of the routine — there are no 'side effects' like changing the random number seed. The routine at address 40000 (or wherever) must assume that the Spectrum ROM — so in other words my USR command is exactly the same as the normal USR function.

The machine code necessary to create this new command is very short, and may be placed at absolutely any address in RAM. To activate it, however, you must *not* CALL it in the usual way. Instead you must activate it by loading the system variable VECTOR (address 5CB7, or decimal 23735) with the address of the label START in my program — that's all you need to do.

Having typed in the machine code to any suitable address (remembering to fill in the address that's labelled SUB_ADDR in the listing, and having POKEd the system variable VECTOR with the address of the label START) you'll find that BASIC has now been extended. Try this Basic program to see:

```
10 INPUT X
20 PRINT X
30 IF X=0 THEN USR 3435
40 GO TO 10
```

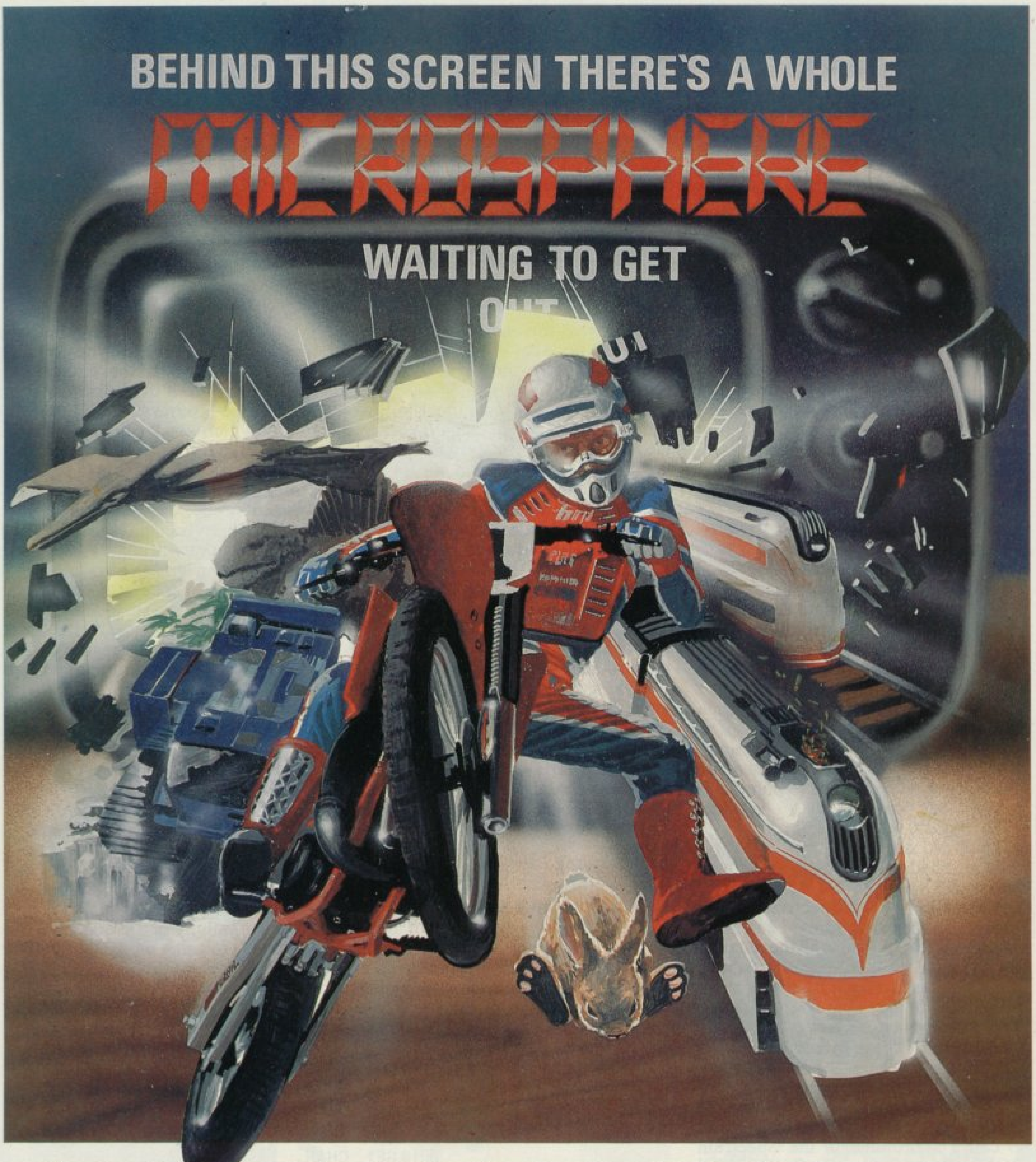
USR 3435 is the subroutine for CLS in the ROM, so you should expect the screen to clear every time you input zero. Does it work?

You'll notice that my machine code listing contains rather a lot of calls to subroutines in the ROMs. Some of these are important for you to know. Address X01F0 is the address you have to jump to if the Basic line is not recognised. Address X05B7 is a subroutine which tests for the end of a Basic line — that is for a colon or a carriage return; if this is not found then a syntax error results. But *more importantly*, if we are only checking the syntax, and not actually carrying out the statement, then this routine will insert the line into a Basic program if it has a line number — or prepare to execute it if not. Control will *not* return from subroutine X05B7 unless we are actually carrying out the statement. Finally, address X05C1 is the address you must jump to when all the work has been carried out.

When extending Basic on your own, it will be useful to know lots of other subroutine addresses in the Spectrum ROM. There are too many of them to list here, so either to grab hold of a copy of *The Complete Spectrum ROM Disassembly* by Dr Ian Logan and Dr Frank O'Hara or, alternatively, get in touch with the ZX Machine Code Users' Club, 37 Stratford Road, Wolverton, Milton Keynes, MK12 5LW.

Incidentally, there's one nice little advantage to using my USR command instead of Uncle Clive's USR function. And that is that with *my* routine the final value of HL' is unimportant, whereas Sinclair Research's routine necessitates that HL' must contain 2758 on return. YS

CODE		ASSEMBLER	COMMENTS
D71800	START	RST X10/ 0018,GET_CHAR	A:= current character — ie. the first character in the Basic line.
FECO		CP "USR"	Is this character 'USR'?
C2F001		JP NZ,X01F0,ERROR	Generate a syntax error if not.
D72000		RST X10/ 0020,NEXT_CHAR	Look at the next character in the line.
D7821C		RST X10/ 1C82,EXPT_1NUM	Check for a numeric expression at this point. Generate a syntax error if not.
CDB705		CALL X05B7,ST_END	Check for the end of the statement. Generate a syntax error if not. If in Edit mode, then exit at this point.
D7991E		RST X10/ 1E99,FIND_INT2	Place the result of the numeric expression in the BC register pair, or generate report code B if out of range.
ED43????		LD (SUB_ADDR),BC	Store this address.
D7		RST X10	
0000	SUB_ADDR	DEFS 02	Call the required subroutine with the Spectrum ROM in place.
C3C105		JP X05C1,LINE_RUN	End of the routine.



* **WHEELIE** (48K Spectrum)
As proud owner of the ultimate racing motorbike, you find yourself in a strange world — a world full of double-decker buses to leap and where even the hedgehogs are out to get you! Your only hope of escape is to find the elusive ghostrider and then beat him in a life-or-death race.
100% machine-code action, keyboard and joystick options, demonstration-mode, and amazing graphics combine to make WHEELIE one of THE games for 1984. . . .only **£5.95**

* **THE TRAIN GAME** (16/48K Spectrum)
The game that starts where model railways left off. Full graphics featuring passenger, goods & express trains; tunnels; stations; turntable bonus games; irate passengers; collisions derailments; and everything else you'd expect from a major railway! just **£5.95**

" . . . an excellent game which is original, well thought-out and full of action" (S. User Nov 83)

"Fun, fun, fun to play . . .
(Home Computing Weekly 27/9/83)

OMNICALC (48K Spectrum)
The best Spectrum spreadsheet you can buy. Written in machine code, to be faster, to give you more space for data, and to include more features, it is guaranteed uncrashable. Complete with comprehensive manual **£9.95**

"If more programs shared the quality of OMNICALC then we might see more Spectrums in offices"
(Home Computing Weekly 3/6/83)

EVOLUTION (48K Spectrum)
Meet Tyrannosaurus Rex, Podopteryx, Brontosaurus and many more fascinating creatures on the journey from the start of life to man. See 3500 million years of evolution compressed into half an hour **£6.95**

CBM 64 and BBC owners — **WHEELIE** and **THE TRAIN GAME** will soon be ready for your micros.

Items marked * are available from selected branches of Boots

Available at good computer shops everywhere, or by mail order from MICROSPHERE, 72, Rosebery Road, London N10 2LA (Tel: 01-883 9411)

The concepts behind computer-aided design (CAD) and, in particular, the delights of three-dimensional plotting, are such that they have usually remained the preserve of 'big' computers. Mathematician and engineer **Damir Skrgatic** *sets out over this issue and the next to prove that usefulness in this area is within the reach of our honest but humble eight-bitters.*

PLOTTING IN 3D/PART ONE
GRAPHICS & MATHEMATICS

For many of its users, maths fascinates firstly for its beauty and elegance in dealing with complex manipulations and relationships, and secondly as a tool which seems almost to possess a mind of its own. Providing that the rules of the 'game' are not broken, it can lead the faithful along a complex path of calculations to what sometimes turns out to be a quite unexpected solution. (Take, for instance, the supreme examples of relativity and quantum mechanics, where results obtained through mathematics contradict everyday common sense!)

But your author must admit that, until the advent of the microcomputer, this fascination with maths proved less than faithful. All too often when trying to solve a problem using pen and paper, the calculations became too complicated to continue. It was like fighting with an octopus — you escape one set of arms only to fall foul of the next!

But how can microcomputers help? Well, even the cheapest available are programmed to carry out mathematical calculations at speeds far beyond our imagination. Thousands of repetitive calculations are easily performed on the execution of a single line of a Basic program. Programs can be written not only to perform these calculations but also to make 'logical' decisions about which formulae should be selected, dependent on interaction between the user and the machine.

Nowhere else are these advantages more useful than in the field of computer graphics. However, the demands on computer power (both speed and storage capacity) are enormous and this is why graphics programs used for computer aided design (CAD) have usually had to run on the world's most advanced machines — typically, 32- and 48-bit machines with ultra-fast parallel processors. These machine code programs are also assisted by special hardware for fast number crunching and display processing.

No wonder then the scarcity of graphics programs written for eight-bit machines in general and home computers in particular. But although CAD seems far beyond their power capability, various programs for fun and education *have* been devised and our articles will attempt to describe some of the concepts used in 3D plotting.

PROJECT OUTLINE

The emphasis here will be on modelling shapes and objects using mathematical functions rather than the plotting of individual points in space. With the choice of a suitable system of co-ordinates, geometrical models can easily be rotated, translated or scaled in size without the need for matrix transformations (see Part 2 in our next issue). This rather unusual approach can produce quite a variety of symmetric figures in perspective.

Anticipating three different kinds of reader, it seemed best to include a reasonably self-contained 3D plotting program with full listing. Maths wizards with Basic experience will probably 'eat their way' through the routines and so also may the reader who is not interested in the maths and who merely wishes to key in the program and have some fun following the examples given. Having dealt with these two extremes, the rest of the has been written from the point of view of those wishing to learn, use and further develop the main concept used in the program and examples. (By the way, the full derivation of the mathematical formulae has been extracted from the main text and dumped in the Appendix, together with a suggested list for further reading.) Let's now enter the main program and, once that's been successfully achieved, try some of the parameter examples shown.

LOOKING INTO PERSPECTIVE

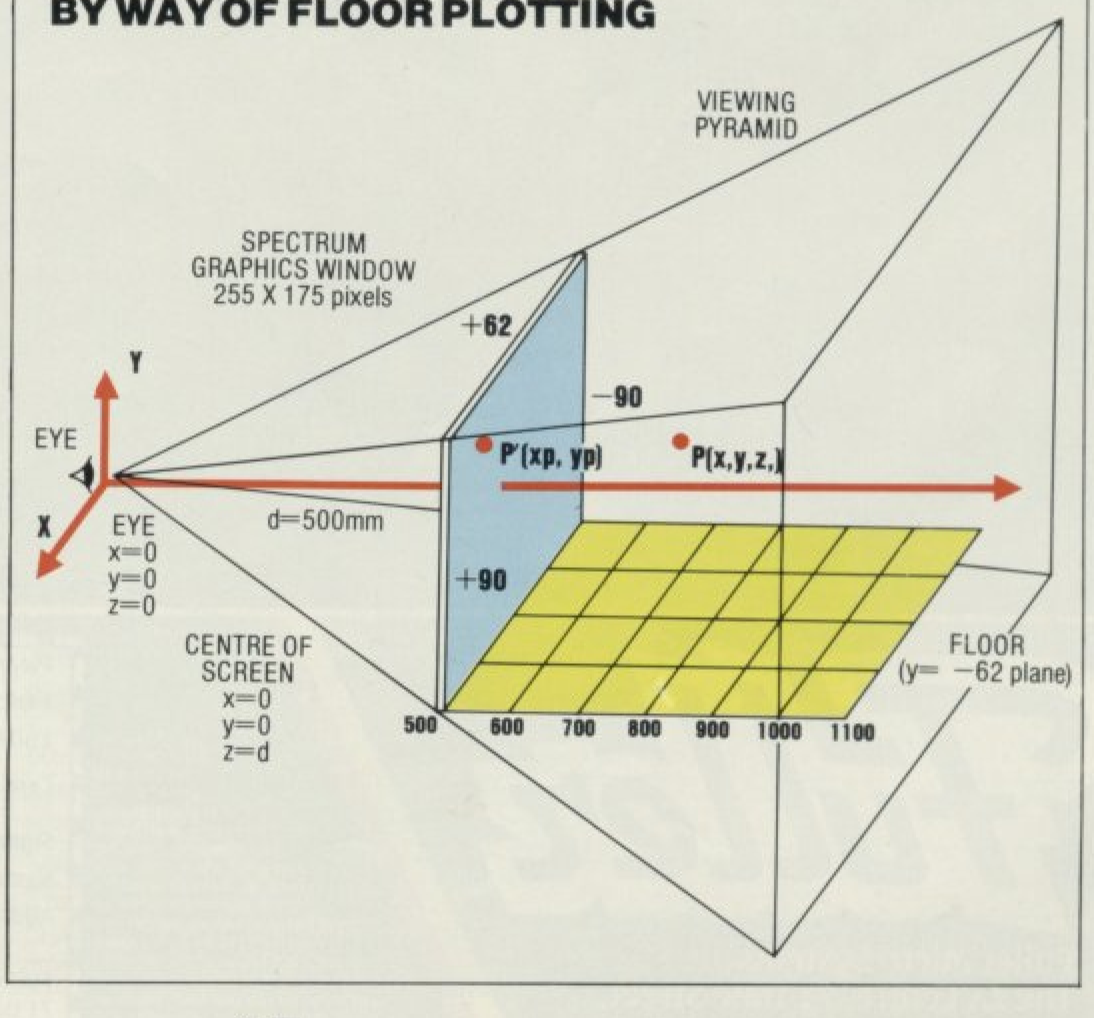
Perspective projection can now be introduced by way of the 'floor' plotting subroutine at line 520; it's also illustrated by further examples shown in the *Perspective Projection* diagram.

There are a number of basic assumptions made at this point:

1. The origin of the rectangular coordinate system is in the observer's eye.

2. All distances are expressed in millimetres (other units can easily be

PERSPECTIVE PROJECTION BY WAY OF FLOOR PLOTTING



Transform your Spectrum now!



Transform your Spectrum now with the FDS Keyboard, the best selling Spectrum keyboard in the U.K. The Fuller FDS Keyboard with its stylish, slim-line appearance gives you the look and feel of a real microcomputer keyboard and helps you enter data with lightening fast accuracy.

The keyboard contains all the graphic characters of your ZX Spectrum plus additional function keys. It has 4 cursor control keys, an auto rub-out key, a separate key for full stop and comma, a full length space bar, shift keys either side and 2 function keys for direct entry into green and red E modes.

The microdrive is easily adapted to go inside along with the power supply. Fixing is simplicity itself, no soldering or technical knowledge is required. For the user who is reluctant to install his Spectrum circuit board inside the FDS, a buffer is available (£8.75 + 80p p&p) which simply plugs into the expansion port and connects directly to the FDS Keyboard, allowing the whole cased Spectrum to be installed inside.

£49.95 + £2.50 p&p

Fuller MICRO SYSTEMS

Fuller Micro Systems
The ZX Centre, 71 Dale Street,
Liverpool 2. Tel: 051-236 6109.

At Fuller Micro Systems we strive to maintain the highest quality in both research and development, ensuring our products are the very best. We have now built a brand new mail order department to ensure our service is also the very best. Our stock includes a full range of components and kits that will make the most of your Spectrum; all our products enjoy an excellent reputation for reliability and carry a world-wide one year guarantee.

Please supply me with............ FDS Keyboard(s).
I enclose a cheque/PO payable to
Fuller Micro Systems for ....or debit my credit card.
Card No. ☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐ ☐Barclaycard ☐Access Card
Signature ....
Name ....
Address ....
....
Fuller Micro Systems, The ZX Centre,
71 Dale Street, Liverpool 2. Tel: 051-236 6109

adopted as long as they are used consistently throughout).

3. Millimetres (mm) are converted to pixels only in the lines of the program which use DRAW and PLOT statements.

4. The number of pixels/mm (p) for the Spectrum graphics window on a 12-inch TV is assumed to be equal to 1.41 (255/180 = 1.41). This nominal value is used throughout.

5. The screen window on your TV should first be plotted using the subroutine at line 450, and then measured with a ruler so that the real size in millimetres can be entered in line 50. If your screen is around 12 inches then line 50 values are correct and program execution can be continued by pressing the CONTinue key. The centre of the screen is at: x = 0, y = 0, and z = d = 500mm (ie. with your eye looking straight at the screen from the distance of half a metre).

6. The object can be placed anywhere within the viewing pyramid which is solely determined by the ratio of the screen size and the eye-screen distance (d). A large screen and a short eye-screen distance will result in the rapidly diverging pyramid, ie. similar to a wide aperture in a photographic lens. Line 290 to 360 limit or 'clip' the parts of the 3D object which fall outside the viewing pyramid.

7. The general conditions to be met for the point to lie within the viewing pyramid are (see the Appendix for a fuller explanation, equation 2a):

$$-z*w/d<=x<=z*w/d$$

(where w is half of the screen window width in millimetres) and:

$$-z*h/d<=y<=z*h/d$$

(where h is half of the screen window height in millimetres).

The values in the program (lines 290 to 360) correspond to the viewing pyramid of d = 500mm, and the Spectrum graphics window on a 12-inch TV.

8. The clipping conditions in practice are slightly more complicated due to the addition of dx and dy which are incremental (vector) components corresponding to x and y increments used in the Spectrum DRAW statement. This Spectrum instruction is slightly non-standard in that it only determines the length and direction of a line (what's known as a vector in maths notation) while the starting point of the vector is the pixel occupied by the last PLOT or DRAW statement. Thus, every DRAW statement must be preceded by a PLOT statement which sets the starting point.

9. Perspective projection of a point, P(x,y,z), on to the screen, P(xp, yp), is based on a simple relationship between the similar triangles, Odxp and Ozx, as shown in *Figure 1* in the Appendix. After simple manipulation:

$$xp = x * d/z \quad (1)$$

and:

THE 3D PLOTTING PROGRAM

COMMENT	LISTING
Goes to the 'draw screen window' routine at line 450.	`40 GO SUB 450: STOP`
Prints the drawing parameters on-screen.	`50 PRINT AT 21 13;"y=-62mm";AT 11,26;"x=90mm";AT 0,13;"y=62mm";AT 11,0;"x=-90mm"`
Define the variables, p and d.	`60 LET p=1.41` `70 LET d=500`

Collect the user-defined parameters and print up a flashing message to tell the user that the necessary calculations are being undertaken by the computer.	`80 GO SUB 520`
	`90 BEEP 1,10: INPUT AT 0,0;"Input distance of object from   eye in mm (z0)=";z0`
	`100 INPUT AT 0,0;"Input 'radius' of object in mm  (R)=";R`
	`110 INPUT AT 0,0;"Input horizontal offset from    center of screen in mm (x0)=";x0`
	`120 INPUT AT 0,0;"Input vertical offset from       center of screen mm (y0)=";y0`
	`130 INPUT AT 0,0;"Input no. of vertical sections  required (sb)=";sb`
	`135 IF sb=0 THEN  PRINT #1; FLASH 1;"NO. OF VERTICAL SECTIONS CAN'T=0": PAUSE 1: PAUSE 150: GO TO 130`
	`140 INPUT AT 0,0;"Input no. of sides of each       section (See Text) (sa)=";sa`
	`145 IF sa=0 THEN  PRINT #1; FLASH 1;"  NUMBER OF SIDES CAN'T=0      ": PAUSE 1: PAUSE 150: GO TO 140`
	`150 INPUT AT 0,0;"Input start longitude in degrees(See Text)  (b0)=";b0`
	`160 INPUT AT 0,0;"Input start latitude in degrees (See Text) (a0)=";a0`
	`163 PRINT #1; FLASH 1;"    CALCULATING PLEASE WAIT     "`

Set the longitude and latitude increments in degrees.	`170 LET db=360/sb` `180 LET da=360/sa`
Set up a FOR-NEXT loop, which is eventually returned by lines 420 and 430.	`190 FOR b=b0 TO 360+b0 STEP db` `200 FOR a=a0 TO 360+a0 STEP da`

Convert the spherical co-ordinates (R,a,b) to cartesian co-ordinates (x,y,z).	`220 LET x=R*COS (PI/180*a)*COS (PI/180*b)+x0`
	`230 LET y=R*SIN (PI/180*a)+y0`
	`240 LET z=R*COS (PI/180*a)*SIN (PI/180*b)+z0`
	`250 LET dx=R*COS (PI/180*b)*(COS (PI/180*(a+da))-COS (PI/180*a))`
	`260 LET dy=R*(SIN (PI/180*(a+da))-SIN (PI/180*a))`
	`270 LET dz=R*SIN (PI/180*b)*(COS (PI/180*(a+da))-COS (PI/180*a))`

Carry out the clipping outside the viewing pyramid.	`290 IF x>z/5.6 THEN  LET x=z/5.6`
	`300 IF (x+dx)>z/5.6 THEN  LET dx=z/5.6-x`
	`310 IF y>z/8.3 THEN  LET y=z/8.3`
	`320 IF (y+dy)>z/8.3 THEN  LET dy=z/8.3-y`
	`330 IF x<-z/5.6 THEN  LET x=-z/5.6`
	`340 IF (x+dx)<-z/5.6 THEN  LET dx=-z/5.6-x`
	`350 IF y<-z/9 THEN  LET y=-z/9`
	`360 IF (y+dy)<-z/9 THEN  LET dy=-z/9-y`

Sets the perspective projections for x, y and z.	`370 LET xp=x*d/z: LET yp=y*d/z`
Plot the projections in pixels (p=pixels/mm).	`380 PLOT xp*p+128,yp*p+87` `383 LET dk=d/(z+dz)-d/z`
Sets the vector perspective projections.	`390 LET dxp=dx*d/z+x*dk: LET dyp=dy*d/z+y*dk`
Sets the colour coded depth.	`400 LET e=ABS ((z-d)/200)`
Draws the vector projections in pixels.	`410 DRAW  INK e;dxp*p,dyp*p`

COMMENT	LISTING
Return the program to the beginning of the 'plotting' routine begun in lines 190 and 200.	`420 NEXT a` `430 NEXT b`
Returns the program to line 90 to collect more user-defined parameters to draw another object.	`440 GO TO 90`

The routine called at line 40 to draw the screen window.	`450 PLOT 0,0`
	`460 DRAW 255,0`
	`470 DRAW 0,175`
	`480 DRAW -255,0`
	`490 DRAW 0,-175`
	`500 BORDER 5`
	`510 RETURN`

The routine called to plot the floor in perspective (wire-frame plot of the y=−62 plane).	`530 FOR z=d TO d+600 STEP 100` `540 FOR x=-90 TO 90 STEP 22.5`
Sets the perspective projection of x on the screen.	`550 LET xp=x*d/z`
Sets the perspective projection of y on the screen.	`560 LET yp=-62*d/z`
Sets the colour coded depth, ie. z=d, f=0 gives black.	`570 LET f=INT ((z-d)/200)`
Plots the projections xp and yp in pixels; 128 and 87 are pixel co-ordinates of the screen centre.	`580 PLOT xp*p+128,yp*p+87`
Sets the perspective scale factor decrement as a function of dz=70.	`590 LET dk=d/(z+70)-d/z`
Draws perspective projections of a vector dz in pixels.	`600 DRAW INK f;x*dk*p,-62*dk*p`
Resets the vector to 580	`610 PLOT xp*p+128,yp*p+87`
Sets the perspective projection of an increment dx=20.	`620 LET dxp=20*d/z`
Limits the draw vector to within the screen window.	`630 IF x+22.5>90 THEN LET dxp=0`
Draws the dx projection in pixels.	`640 DRAW INK f;dxp*p,0`
Returns the program for new values of x and y	`650 NEXT x` `660 NEXT z`
Returns to the main program.	`670 RETURN`

$$yp = y * d/z \qquad (1)$$

ie. projections, xp and yp, are inversely proportional to depth, z.

The perspective will now be demonstrated by way of the routine used in the program to plot the 'floor' in a wire-frame fashion. In maths notation, this represents a plot of y = −62mm plane as shown in the *Perspective Projection* diagram.

Looking at the program, line by line, the following can be seen:

Lines 530 and 540 define the size of the plane, ie. z (depth) between 500mm and 1100mm, and x (width) between −90mm and +90mm.

Lines 550 and 560 are direct applications of equations (1), ie. perspective projections of each intersection of 'wires' in the plane, y = −62 = constant.

Line 580 plots projections, xp and yp, expressed in pixels (multiplying with p) and sets the PLOT origin in the middle of the screen window (255 by 175) pixels.

Line 590 defines the decrement of perspective scale factor, dk, as z is incremented by dz = 70. This is necessary in order to work out the projected x and y decrements as a result of an increase in depth, dz.

Line 600 contains the first DRAW statement which draws a line on the screen corresponding to the projection of the increment (vector), dz, through the application of dk.

Line 610 resets the DRAW vector back to 580.

Line 620 converts an increment in x (dx = 20) to its projection, dxp.

Line 640 contains the second DRAW statement and draws a projection corresponding to a dx = 20 increment.

PLANE PLOT SUMMARY

'Wire' cross sections in the 'floor' were obtained through a direct application of equations (1) and the PLOT statement. A DRAW statement is used to draw the projections of dz and dx increments, which are actually vectors since they not only have a length but also a direction in space. In fact, it wouldn't be difficult to plot a 'wire-frame' of any plane. For example, the x = 90 plane can be plotted by substituting y for x in the following program:

```
520 REM 'Right wall, x = 90 plane'
540 FOR y = -62 TO 62 STEP 12
550 LET yp = y * d/z
560 LET xp = 90 * d/z
...
...
...
```

50

PARAMETER EXAMPLES FOR THE 3D PLOTTING PROGRAM

	Ceiling lamp, with shade and flex		Window frame on . . .		Edge between . . .						door	decor on . . .	
			back wall	right wall	left wall	left and back wall	right and back wall	back wall and ceiling	left wall and ceiling	right wall and ceiling		ceiling	wall
z0	900	900	1100	800	600	1100	1100	1100	800	800	1000	900	700
R	15	3.5	25	25	30	62	62	90	300	300	50	90	40
x0	0	0	0	90	−90	−90	90	0	−90	90	−90	0	90
y0	40	58.5	0	0	0	0	0	62	62	62	−27*	62	0
sb	3	1	1	1	1	1	1	1	1	1	1	10	1
sa	8	2	4	4	4	2	2	2	2	2	4	2	4
b0	45	0	0	90	90	0	0	0	90	90	90	0	90
a0	0	90	45	45	45	90	90	0	0	0	45	0	0

*(−67+45)



Complete your computer
with the sensational KELWOOD SOUND POWER BASE
has all the 'basic bits' that Sinclair left off.
● Fully adjustable sound amplification
● ON-OFF re-set switch
● LOAD/SAVE switch
● Angled stand
● Does not take up any extra space ● All wires included ● Saves wear and tear on connections ● Gives distinct click with each key depression ● Amplifies all sound on program and on cassette ● Does not need batteries ● Does not interfere with inside of Spectrum or any other add-ons ● No soldering required, simply plugs in
REF SPBI £19.90
Standard Power Base without sound
SPECTRUM REF PBSI £13.50
ZX81 REF PBZXI £13.00
GET REAL SOUND FROM YOUR SPECTRUM
KELWOOD add-its for Sinclair addicts
BACKPACKS
Six versions of this indispensable unit complete your computer. All have SAVE/LOAD and ON/OFF switching and a housing for the transformer. Compatible with interface one. Ideal for use with cased keyboards
Ref BP1 – Fully variable sound amp – 3 x 13 amp sockets, switch and neon indicator – cable and plug £27.50
Ref. BP2 – as BP1 but no sockets, cable and plugs. Has room to fix your own 'Duraplug' sockets £19.50
Ref. BP3 – as BP1 but no sound £21.50
Ref. BP4 – as BP1 but no sound or sockets £13.05
Ref. BP5 – ZX81 version inc. sockets, mains switch neon, cable and plug £20.85
Ref. BP6 as BP5 but no cable or sockets £12.85
WOBBLE STOPPER
STOPS ZX81 RAM-PACK WOBBLE
Flat: Ref SW £5.25. Tilted: Ref STW £6.25
Extra long for printer – Flat: Ref LW £5.75. Tilted: Ref LTW £6.75
COOL-IT UNIT
Takes the heat out of your computer. Lets your Spectrum or ZX81 run much cooler.
Separate Cool-it Unit
ZX81 Ref ZXC £11.65 Spectrum Ref SC £11.85
Cool-it units built into other Kelwood add-its
Power Base (Spectrum) Ref PBSIC £18.55 ● Power Base (ZX81) Ref PBZXIC £19.35 ● Sound Power Base (Spectrum only) Ref SPBIC £25.00 ● Back Pack 1 Ref BP1C £32.55 ● Back Pack 2 Ref BP2C £24.55 ● Back Pack 3 Ref BP3C £26.55 ● Back Pack 4 Ref BP4C £18.60 ● Back Pack 5 Ref BP5C £26.90 ● Back Pack 6 Ref BP6C £18.90 ● Wearsaver (Spectrum) Ref WSSC £16.00 ● Wearsaver (ZX81) Ref WSZX £15.80
WEAR SAVERS
● Cuts out wear and tear on plugs and sockets ● Switching for SAVE/LOAD and 9 volts ON/OFF. ● Simply plugs in
Spectrum Ref WSS £9.95
ZX81 Ref WSZX £9.75
MICROSTATION
The compact way to work ● Room for all essential peripherals ● Ideal for use in restricted spaces
● Allows the computer etc to be moved around with the minimum of effort
Ref. MS £7.50
KELWOOD almost A WIRELESS WORKSTATION
● Mains ON/OFF switch ● 3x13 amp sockets inside ● Supreme quality ● All wires stow away ● Carrying handle ● Easily stored ● Sturdily built in stove enamelled steel ● Ideal for schools ● Suitable for most micros including Sinclair and BBC ● Acres of room for all peripherals
Desk top model
Ref CW1 £49.00
Legs Ref LCW £11.00
Name
Address
CREDIT CARD HOTLINE TEL: (0709) 63242 8am-10pm
QUANTITY
REF
AMOUNT
Items under £10 plus 60p P&P
Items over £10 plus 95p P&P
Workstations plus £3 P&P
TOTAL ENCLOSED
Access
Barclaycard No
BARCLAYCARD VISA
KELWOOD COMPUTER CASES Downs Row, Moorgate, Rotherham PERSONAL CALLERS WELCOME
All prices include VAT
Y.S.

```
600 DRAW INK f; 90* dk * p, y * dk * p
...
...
...
620 LET dyp = 8 * d/z
630 IF y + 12 > 62 THEN LET dyp = 0
640 DRAW INK f; 0, dyp * p
650 NEXT y
660 NEXT z
```

Similarly, the the z = 1100 plane could be plotted using the following program:

```
520 REM 'Back wall'
525 LET z = 1110
530 FOR y = -62 TO 62 STEP 12
...
...
...
560 LET yp = y * d/z
...
...
...
590 LET dyp = 12 * d/z
595 IF y + 12 > 62 THEN LET dyp = 0
600 DRAW INK f; 0, dyp * p
...
...
...
650 NEXT x
660 NEXT y
```

Alternatively, all three planes could be combined into a 'bottom-right corner of a room' by writing a main program with the three examples used as subroutines.

A side view of a 'house with three floors' can easily be generated from the original floor, y = -62, by adding the following lines:

```
517 REM 'Three floors'
520 FOR y = -62 TO 62 STEP 50
...
...
...
560 LET yp = y * d/z
...
...
...
600 DRAW INK f; x * dk * p, y * dk * p
...
...
...
665 NEXT y
```

The analogy could be taken even further, eg. for plotting a plane under any orientation and, with a combination of several planes, one could generate 'boxes' in space. It's obvious, though, that the programs would become unmanageably complex.

Based on experience so far, the 'plane plotting routine' is best recommended only for the following three cases:
1. Single 'floor' or 'wall'
2. Multiple parallel 'floors' or 'walls'
3. Single or multiple 'parallel' surfaces of which one example is given here (generated from the original 'floor' routine):

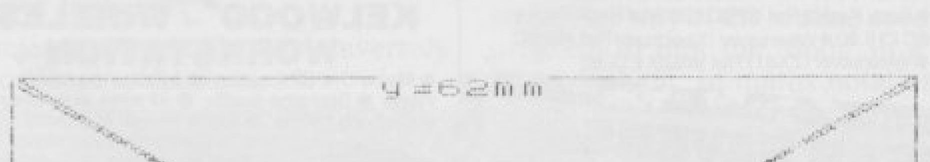
```
520 REM 'Sine surface 'roof', y = +40'
530 FOR z = d TO d + 600 STEP 100
540 FOR x= -90 TO 90 STEP 22.5
555 LET y = 40 + 10 * SIN(x/10)
560 LET yp = y * d/z
...
...
...
600 DRAW INK f; x * dk * p, y * dk * p
...
...
...
625 LET dy = 10 * (SIN(( x + 20)/10)-
    SIN(x/10))
628 LET dyp = dy * d/z
...
...
...
635 IF y + dy > 62 THEN LET dyp = 0
...
...
...
640 DRAW INK f; dxp * p, dyp * p
```

IN PART TWO

The generation of 3D objects using spherical co-ordinates will explain the role of parameters in INPUT statements — some examples of which appear in Part 1.

Suggestions for further reading

1. WM Newman, RF Sproull, *Principles of Interactive Computer Graphics,* McGraw-Hill, 1979
2. Nick Hampshire, *Spectrum Graphics,* Duckworth, 1982
3. 'Spectrum 3D Rotator', page 80, *Your Computer*, July 1983



Using the 3D Plotting program and the tabulated parameter examples, this is what you should end up with on the screen. Why not have a go with some of your own parameters and see what you come up with . . .

APPENDIX

PERSPECTIVE TRANSFORMATION

POINT PROJECTIONS

$xp/d = x/z$

$xp = x * d/z$ . . . . (1a)

$x/z <= w/d$

$x <= z * w/d$ . . . . (2a)

$yp/d = y/z$

$yp = y * d/z$ . . . . (1a)

$y/z <= h/d$

$y <= z * h/d$ . . . . (2a)

Point transformation (*Figure 1*) is derived from similar triangles, Odxp and Ozx, for x; this can also be seen for y. This transformation is used to plot points using the PLOT statement. The viewing pyramid determines the 3D space in the front and back of the screen which can be used without clipping (see equations 2a).

VECTOR PROJECTIONS

$dxp = dx * d/z + x * dk$

$dyp = dy * d/z + y * dk$ . . . . (3a)

The combined (total) vector projection is shown in *Figure 2*.

'Vector transformation' is used with the Spectrum DRAW statement in order to draw projections of increments (vectors) dx, dy and dz.

1. Projections of dx and dy are derived in the same manner as projections of individual points (see *Figure 3*).

$dxp = dx * d/z$

$dyp = dy * d/z$ . . . . 4a)

2. The projection of dz is derived via dk. dk is defined as the decrement of the perspective scale factor ($k = d/z$) due to (only) an increase of z by dz:

$dk = (d/(z+dz)) - dz$ . . . . (5a)

Referring to *Figure 4* and applying the standard point transformation (see equations 1a) for the points P1 (x1,y1,z) and P2(x1,y1, z + dz), the following can be derived:

$xp1 = x1 * d/z,\ xp2 = x1 * d/(z + dz)$

$yp1 = y1 * d/z,\ yp2 = y1 * d/(z + dz)$ . . . . (6a)

Since, $dxp = xp2 - xp1$
$= x1 * d/(z + dz) - x1 * d/z$
$= x1 * (d/(z + dz) - d/z)$
$= x1 * dk$

ie. dxp (due to only dz) = $x * dk$
dyp (due to only dz) = $y * dk$ . . . . (7a)

The combined effect of all three increments, dx, dy and dz, is therefore a sum of all component projections (see equations 3a and *Figure 2*). The equations are now repeated for convenience:

$dxp\ (tot) = dx * d/z + x * dk$

$dyp\ (tot) = dy * d/z + y * dk$

due to dx or dy — due to dz only

Figure 1
POINT TRANSFORMATION

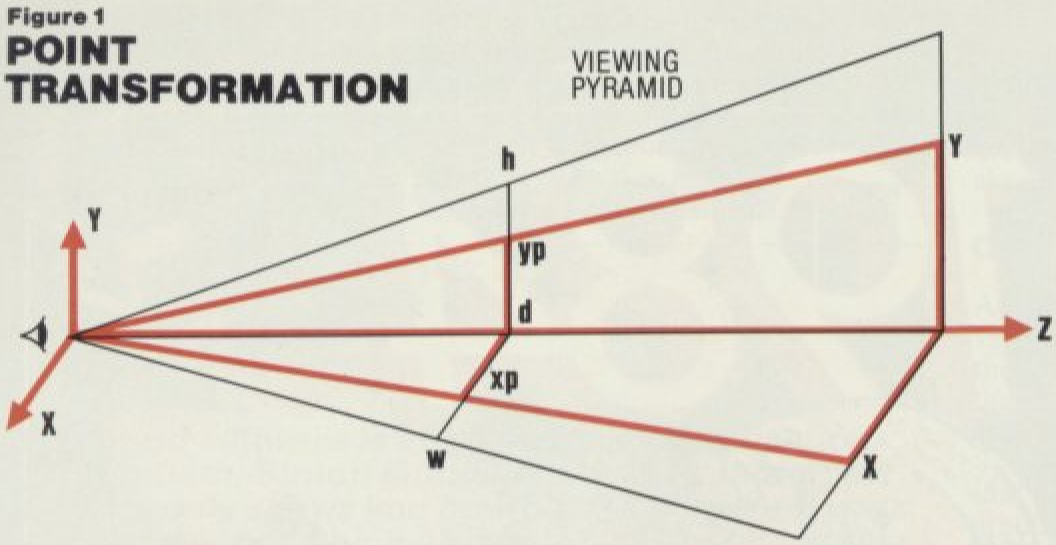


Figure 2
VECTOR PROJECTIONS

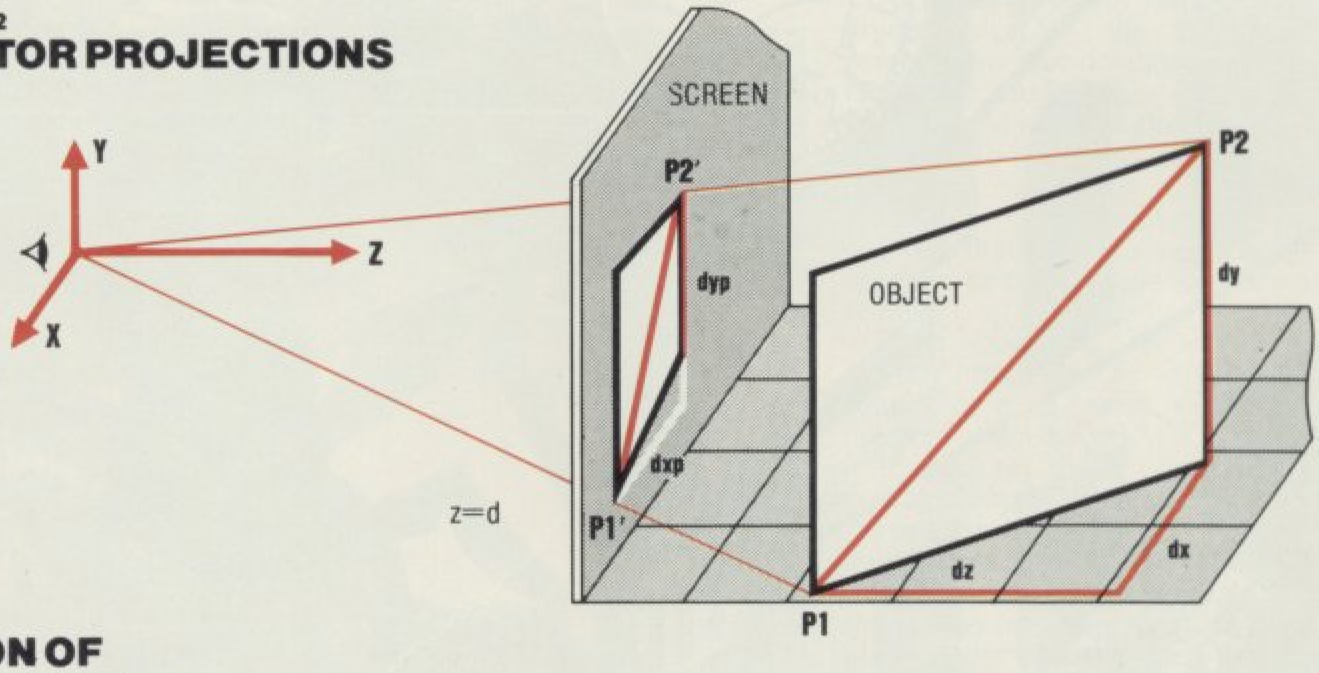


Figure
PROJECTION OF INDIVIDUAL POINTS

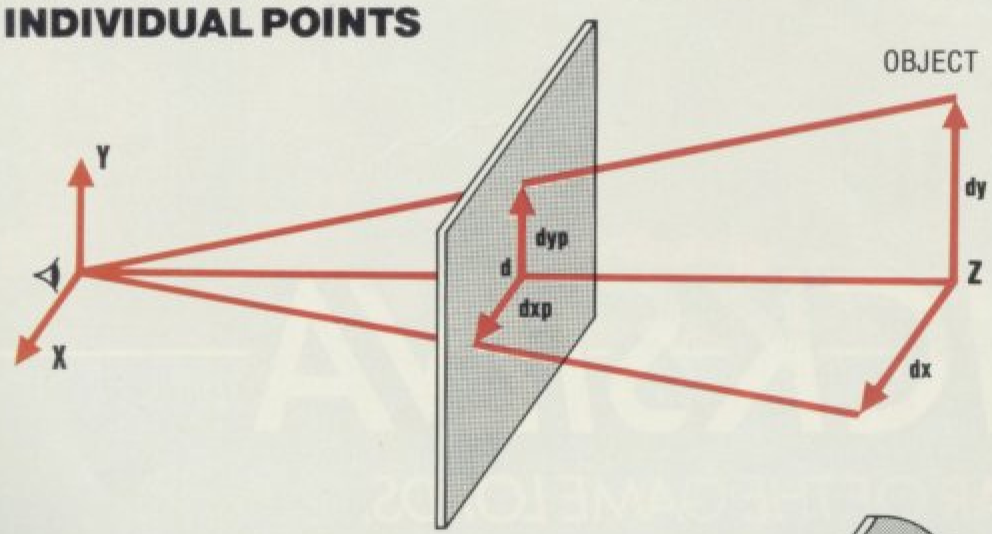
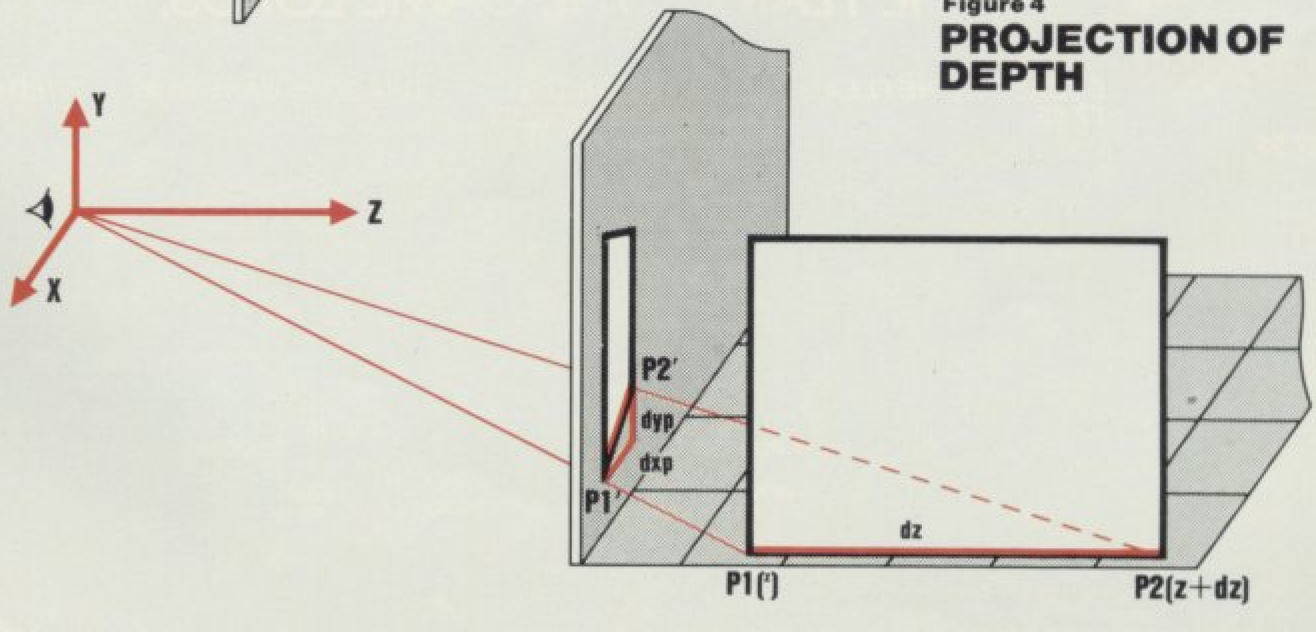


Figure 4
PROJECTION OF DEPTH



MARK WATKINSON


1984
QUICKSILVA

ADDING ZIP!

PART 1

Designing good software is all a matter of organisation. Starting this month,* Simon Goodwin *begins a three-part feature that not only shows you how best to construct your Basic programs — but which also gives you an excellent compiler program into the bargain!



MICHAEL BROWNLOW

Starting this month, we set out to illustrate the design thinking which goes into a complicated program, and to develop a useful software tool in the process. We'll list and explain ZIP, a powerful Basic compiler for the 48K Spectrum, describing how it was designed. The compiler produces very fast code — often 100 times faster than interpreted Basic. It's also very easy to use, with unusually good error-trapping, and yet it's a fairly short program — about 20K of commented Basic and 1.5K of machine code.

Basic programmers have to put up with a lot of criticism from academics and whizz-kids. They might not actually say that too much Basic will make you go blind, but they certainly imply it! This program shows that you *can* write intricate, powerful programs in Basic and there's no reason why they shouldn't be concise and readable too.

COMPILERS

A compiler is a program which converts Basic into machine code. Ideally, the machine code performs in exactly the same way as the Basic — but much more quickly. Anyone who's had a Spectrum for a while will have noticed that all of the fastest, flashiest programs are written in machine code rather than Basic.

At the heart of any micro is a fairly crude gadget — the processor. In essence this can do three things — it can move small values in memory; it can add and subtract them; and it can 'JUMP' to look at different areas depending upon the results of its arithmetic. The processor's saving grace is that it works very fast — at roughly half a million operations a second.

Even though the processor can't directly multiply or divide, make sounds or print messages (plus hundreds of other things), it is possible for it to perform those operations with combinations of the simple steps which the processor *can* handle. For instance, multiplication can be done by repeatedly adding, printing can be carried out by moving patterns to the display memory, and so on. When you turn on your Spectrum, the thousands of instructions in the 16K ROM read commands from the key switches and perform appropriate actions — step by tiny step.

The 16K ROM is called an interpreter, because (just like a human translator) it converts words in one language — Basic — into another language, machine code. This involves two operations; first the instruction must be recognised, then it must be acted upon. The snag is that recognising simple Basic commands often takes much longer than performing the required action. An interpreter never 'learns by its mistakes'. You can run this program:

```
10 FOR I=1 TO 1000
20 LET X=2+2
30 NEXT I
```

and the value of X will be worked out as slowly the thousandth time as it was the first. Each time Basic looks through the line to make sure it isn't anything nonsensical (like LET 7="WALLY"), it finds out where it keeps the value of 'X' (by searching a list), then it finds the binary form of the number '2'. Computers use binary arithmetic, unlike the decimal system which caught on among humans a couple of millennia ago. Values have to be converted accordingly before they can be accepted or displayed.

Next, Basic makes a note that it will need to do an 'add' (once it has two num-

THE INDUSTRY STANDARD INTERFACE

FULLY MICRODRIVE COMPATIBLE



From £39.99 Inclusive

SEE US ON STAND E24

AT THE MIDLAND COMPUTER FAIR – BIRMINGHAM.



At last you can have real print performance from your ZX Spectrum with the Kempston Centronics Interfaces. As the Interfaces allow you to link your ZX Spectrum to any of a vast number of printers with an industry standard centronics input, you can choose the printer that suits *your* needs – from high speed dot matrix to professional letter quality printers.

We recommend Epsons, NEC, TEC, Seikosha, OKI Microline, Tandy GP115, Star DP 510, Shinwa, Brother HR15, etc.

NEW

INTERFACE E – ONLY £55.00
Simply plug in and it's ready to use. All operating commands are held in an EPROM so LLIST, LPRINT and COPY can be used at any time without using up valuable user RAM. COPY will allow the reproduction of high resolution graphics with Epson (or derivatives) and Seikosha 80, 100 and 250 Series printers. Print width selection from 32 characters to full width depending on printer used.

INTERFACE S – ONLY £39.99
Visually identical to Interface E but without the EPROM, Interface S also recognises the LLIST & LPRINT commands and will allow print width selection from 32 characters to full width.

However, software routines will need to be loaded before use. Full screen dump to reproduce high resolution graphics is also possible and supporting software is supplied to operate this facility with Epson and Seikosha printers. The software routines that are necessary to initialise the interface are held in the printer buffer so valuable user RAM will not be used up. There is a growing range of Business/Utility software that includes these routines. Details available on request.

Either interface simply plugs into the ZX Spectrum expansion port or interface and is supplied fully cased with a one metre ribbon cable which connects to the printer of your choice. Full instructions are included and driving software is supplied with Interface S.

KEMPSTON CENTRONICS INTERFACE COMPATIBLE SOFTWARE UTILITIES.
FOR THE 48K SPECTRUM.

FINANCE MANAGER (OCP) – Menu driven program for all domestic and business accounting applications. **£19.95**

ADDRESS MANAGER (OCP) – Simple index filing system ideal for names, addresses, etc. Various search facilities. **£19.95**

FULL SCREEN EDITOR/ASSEMBLER (OCP) – Allows you to write Z80 assembly code using standard mnemonics on full screen. **£19.95**

WORD PROCESSOR (TASWORD TWO) – (TASMAN) A professional word processor allowing 64 characters per line and incorporating all usual editing facilities. **£13.90**

OMNICALC (MICROSPHERE) – The only spreadsheet written entirely in machine code. The easy and fast way of solving any numerical problem. Ideal for cash flow forecasting to concrete stress analysis. **£9.95**

MASTERFILE (CAMPBELL SYSTEMS) – Filing and retrieval system for domestic or business use. Files can be loaded and saved independently. Microdrive compatible. **£9.95**

NOW WE CAN ALSO SUPPLY YOUR PRINTER.

We've looked at the printers currently available and have selected what we feel is best value for money in dot matrix and daisy wheel printers:

EPSON RX-80 F/T – A dot matrix printer allowing full graphics dumping and a choice of printing styles. Speed: 100 C.P.S.
Price £325 Inc. VAT and Delivery

BROTHER HR15 – A daisywheel printer ideal for letters, mail shots, documents, etc. Many typefaces available by changing daisywheel. Duplication facility but no graphics. Speed: Up to 18 C.P.S.
Price £425 Inc. VAT and Delivery

Interface S
Available from
spectrum COMPUTER CENTRES

WHSMITH

All products direct from:

KEMPSTON MICRO ELECTRONICS LTD

Unit 30 Singer Way, Woburn Road
Industrial Estate, Kempston,
Bedford, MK42 7AF. Tel: (0234) 856633
Telex: 826078 KEMPMI G

All prices include VAT and P. & P.
Overseas orders please add £4.00 P. & P.
Please allow 21 days for delivery of interfaces and software. Printers available within 48 hours.

TRADE ENQUIRIES
WELCOME

For ZX Spectrum

ADDING ZIP!

bers to play with), it finds another '2', laboriously works out the binary form *again*, adds the two numbers using complicated instructions which are designed to handle all cases (the same code would perform 0.0002 + −99999) and puts the result away under the name 'X'.

The Spectrum has used hundreds of simple operations, where four (fetch, fetch, add, store) would have sufficed. A Basic compiler looks at the listing of a program — in the jumbled order which we humans can understand — and converts it into simple steps in the order favoured by the computer. The juggling about, testing and searching are almost eliminated and you end up with a machine code program which works just like the Basic, but faster.

The main problem is that it's hard to alter the machine code — each operation is dependent upon the ones before it. In practice, you must re-translate (or 'compile') the entire program every time you want to make a change. This is a complicated (and hence, slow) task.

There are two points to be noticed here. Firstly, the compiler (the program which does the translations) doesn't have to be very fast, since it will only need to be used infrequently — you can test your programs in 'slow motion' with the interpreter and only compile them when they work. Secondly, a compiler doesn't have to recognise the whole language to be useful. So long as you know which commands are allowed (and they're sensibly chosen) you can tailor your program to suit. After all, it's sure to be easier to use even a cut-down Basic than pure machine code! However, the compiler must be able to spot — and warn you of — instructions that it doesn't recognise, and make it easy for you to go back and change them.

HOW GOOD IS ZIP?

The standard way of testing Basic is to run eight short programs, called Benchmarks. The longer these take to run, the less the Basic can do in a given time. Spectrum Basic is generally considered slow, but the figures in the *Benchmark Table* show that compilation makes it very much faster. The first column shows the time (in seconds) taken for the interpreter to perform a test, then comes the time taken by compiled Basic, then the ratio of the two. ZIP achieves some of its speed by only allowing whole numbers. Benchmark 8 uses complicated arithmetic which ZIP can't compile, so normal Basic beats ZIP hands down in that particular case!

The second table lists the Spectrum Basic words allowed by ZIP. If you use other words, the compiler will display an error message at the appropriate point in the listing and refuse to compile the program.

In the interests of speed and simplicity some Basic words aren't recognised by ZIP. The remainder is still enough to write almost any program, given sufficient effort! In future issues we'll show you how to add features (including Hi-res graphics!) by altering the compiler.

You may use 26 single letter numeric variables and 26 arrays. The usual arithmetic operators '+', '−', '*' and '/' are allowed, but ZIP only stores whole numbers between −32767 and 32767. For example, PRINT 99/10,5/6 displays '9' and '0'. In fact, you can get away with values between 32768 and 65535, so long as you don't try to multiply or divide them (ie. they may be the result of a multiplication).

No decimal arithmetic is allowed, so SIN, COS, RND and so on are obviously not usable. Likewise strings and streams (PRINT# etc) can't be used, althought simple routines can be used to replace RND and INKEYS$ for games. The words in the *Words Allowed Table* are marked with an asterisk if their usage is restricted by ZIP.

RESTRICTIONS

The ZIP CLEAR command doesn't allow a following number. Since the compiled program is machine code it doesn't use the Basic memory areas which CLEAR <number> protects.

The DIM command differs in that all arrays must be dimensioned — but only once, each — and the DIM statement must be followed by a number, not an expression. The compiler uses DIM statements to tell how much memory to reserve for arrays. DIM is only recognised while compiling, so you can't clear an array by re-dimensioning it, as you can in Basic. All arrays are cleared when the program starts or a compiled CLEAR statement is executed.

ZIP speeds up GO TO and GO SUB statements by converting them into direct JUMPs to the instructions concerned. Whereas Basic has to search for lines by number, the compiler knows exactly where each line is stored. This is much quicker, especially in long programs where Basic would have to search through many lines. Even in a 17-line program, a compiled GO TO was found to be 1400 times faster than the original Basic! GO SUB and RETURN are improved almost as much. The flaw of

PROGRAMMER PROFILE

As Simon Goodwin seems to have nestled quite comfortably between the pages of *Your Spectrum* — and you're bound to have the pleasure of his scribblings for some time to come — we thought it might be nice to tell you just a little of the man behind the keyboard. He began his programming career in '79 on the original Apple, but soon progressed through the Video Genie, Dragon, Memotech MTX 512, Atari 800, Commodore 64 and, of course, our own lovable Speccy He was also software designer on Central TV's *Magic Micro Mission*, presenter of Radio Wyvern's *Computer Club* and author of DK'Troniks'*Gold Mine*. And if that's not impressive enough, he also did a stint as a computer-aided design programmer — and if you were watching *Tomorrow's World* the other night, you'll have seen some of his stuff in action.



WORDS ALLOWED BY ZIP COMPILER

ABS	GO SUB*	OVER
AND	GO TO*	PAPER
AT	IF	PAUSE
ATTR	IN	PEEK
BIN	INK	PRINT
BORDER	INPUT	POKE
BRIGHT	INT*	REM
CHR$	INVERSE	RETURN
CLEAR*	LET	RANDOMIZE
CLS	NEXT	SGN
DIM*	NOT	STOP
FLASH	OR	TAB
FOR	OUT	USR

COMPARATIVE BENCHMARK TIMINGS

TEST	DESCRIPTION	ZX BASIC	ZIP	RATIO
BM1	FOR loop	4.88	0.044	110 times
BM2	IF loop	9.02	0.058	155 times
BM3	BM2 + variable arithmetic	21.93	0.770	28 times
BM4	BM2 + number arithmetic	20.68	0.640	32 times
BM5	BM4 + GO SUB/RETURN	25.22	0.660	38 times
BM6	BM5 + FOR loop	62.80	0.910	69 times
BM7	BM6 + array storage	89.96	1.070	84 times
BM8	Maths functions	25.07	CAN'T	0 times

ADDING ZIP!

this approach is that ZIP must know exactly which line a GO TO or GO SUB is aiming at. Lines such as GO TO 100+X or GO TO L can't be compiled.

INPUT reads numbers from the keyboard into arrays or other variables, but ZIP doesn't let you specify items to be printed at the bottom of the screen. If you put more than one variable in an INPUT statement, they will be assigned from the keyboard one by one. The ZIP INPUT routine won't let you enter a number greater than 32767 or less than −32767. The only keys recognised are the digits, DELETE, ENTER and an optional minus sign at the start.

The INT function is ignored by the compiler, since ZIP always uses integer (whole number) arithmetic. However, it's useful to put INT statements in your programs so that they give the same results — whether Basic or compiled. To be sure, always INT the value produced after a division.

FILLING YOUR 48K

The ZIP compiler places more demands on your RAM than most programs. The panel shows the memory map used — the way that ZIP shares out the 48K between Basic, machine code, and the various tables used during compilation.

The memory map is one of the most important features of the compiler — it has a crucial bearing on the speed, size and power of the program. The map really consists of two independent sections. One is used by the Sinclair ROM, the other by the compiler. The dividing line is the point labelled 'SPECTRUM BASIC AREA'. From that point downwards the map is essentially the normal Spectrum one. The only difference is the division of the program area into two sections.

The Basic of the ZIP compiler uses about 20K of memory. The listing which we will publish next month uses line numbers from 5000 upwards, leaving lines 1 to 4999 for the Basic to be compiled. These should be plenty — remember that ZIP bans computed line references, so it should be easy to renumber a program before compilation. We could have chosen to compile from tape, gaining memory space in the process, but we would then have lost the advantage of being able to RUN the original Basic or the compiled code at will, without loading or saving.

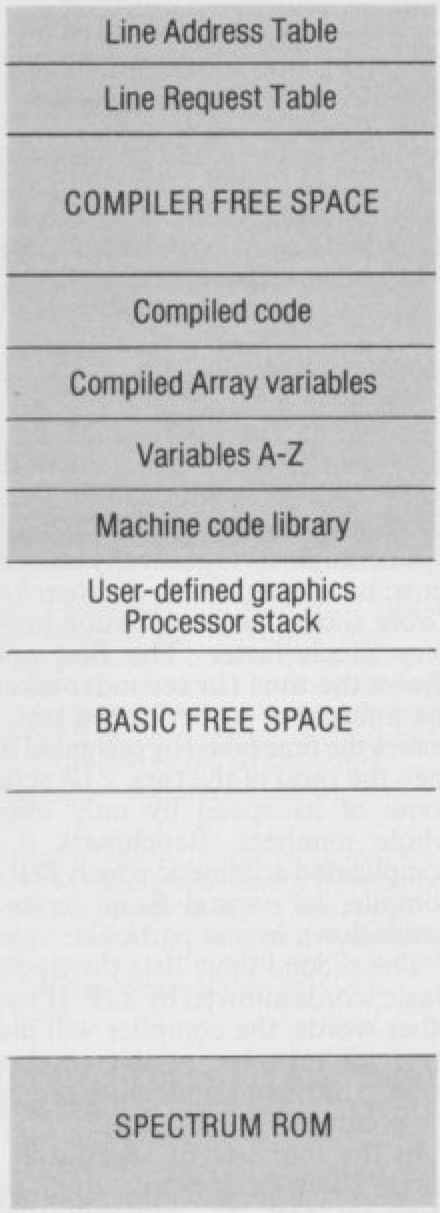
OVER THE TOP

The top of the Basic area is set, as usual, by a CLEAR statement. Depending upon the relative sizes of the Basic and machine code, the top can be adjusted to give more memory to Basic or ZIP. There's about 21K to be shared out between the two. Programs with lots of arrays and comments usually end up shorter in machine code than they were in Basic. Compiled programs packed with comparisons, loops and arithmetic will probably be longer than Basic.

It's useful to be able to save and run a compiled program on its own. ZIP automatically displays the required SAVE statement which should be used to store the machine code, library and variables on tape (notice how they are all stored one after another). User-defined graphics can optionally be stored as part of the same block.

Any areas of memory is best used when there are only two 'dynamic' (growing or shrinking) tables. If you have more than two dynamic areas, it's difficult to know when you have run out of space since some of them must grow in the same direction. Collisions have to be sorted out by moving an entire table out of the way. ZIP uses dynamic areas to store compiled code, and cross-references with the original Basic. We can accommodate these neatly by making one (the code) grow 'upwards' from the bottom of memory, and growing the cross-references down from the top.

ZIP COMPILER MEMORY MAP



HOW ZIP CONVERTS A PROGRAM TO MACHINE CODE

Basic (63 bytes)

```
100 LET K=0
110 LET K=K+1
120 IF K<1000 THEN GO TO 110
130 STOP
```

Intermediate code (30 bytes)

```
FETCH  0
STORE  K
FETCH  K
FETCH  1
ADD
STORE  K
FETCH  K
FETCH  1000
LESS?
IF (false go to next line)
GO TO  110
STOP
```

Machine code (49 bytes)

```
L100   LD     HL,0
       LD     (VARIABLE K),HL
L110   LD     HL,(VARIABLE K)
       PUSH   HL
       LD     HL,1
       POP    DE
       ADD    HL,DE
       LD     (VARIABLE K),HL
L120   LD     HL,(VARIABLE K)
       PUSH   HL
       LD     HL,1000
       POP    DE
       OR     A
       SBC    HL,DE
       LD     HL,0
       JR     C,IF
       JR     Z,IF
       INC    HL
IF     LD     A,H
       OR     L
       JP     Z,L130
GOTO   JP     L110
L130   RST    8
       DEFB   STOP-ERROR
```

ZIP INSTRUCTIONS

VAR FETCH	NUMBER FETCH	ARRAY FETCH	GO TO
GO SUB	RETURN	FOR	NEXT
INPUT *	CLS *	VAR STORE	POKE
ARRAY STORE	OUT	MULTIPLY *	ADD
SUBTRACT	DIVIDE *	BORDER *	INK *
PAPER *	FLASH *	BRIGHT *	INVERSE
OVER	PR-STRING *	PR-NUMBER *	PR-INK
PR-PAPER	PR-FLASH	PR-BRIGHT	PR-INVERSE
PR-OVER	PR-AT	PR-TAB	PEEK
USR	IN	SGN	ABS
NOT	LESS	LESS/EQUAL	GREATER
GREATER/EQUAL	UNEQUAL	PR-CHR$ *	IF
CLEAR *	ATTR *	EQUAL	PR-OPEN *
STOP *	OR	AND	NEGATE
RANDOMIZE	PAUSE *	(PLOT *)	(DRAW *)

ADDING ZIP!

Sinclair Basic uses a plethora of dynamic areas, which is one of the reasons it's rather slow. For a full explanation of the way the Spectrum uses memory, read chapter 24 of the manual.

PASSING THE BUCK

ZIP is a 'three pass' compiler, meaning that it scans three times through the program to be compiled. In theory you can compile Basic with two passes, one to generate code and the other to 'patch' GO TOs and GO SUBs which jump forward in the program. However, these can't be turned into direct JUMPs immediately, since they refer to lines which have not yet been compiled (so the compiler doesn't know what their addresses will be).

ZIP uses an extra pass before any code is generated: a 'quick scan' through the program, checking for the most common and obvious mistakes. The program is listed onto the screen as it's processed, and helpful messages appear at the position of each error. The main idea of this pass is to quickly and accurately report simple errors — if you've made a trivial mistake at the end of the program, you don't have to wait for almost all of the code to be generated before you discover the problem.

Some, more complicated, errors are only found during Pass 2, when the compiler examines your program in detail. Unlike other Spectrum compilers, ZIP doesn't give up at the first error it finds — it carries on and lists all of the errors before stopping.

By now you may have smelt a rat. It's useful to have some general information about a program, before code generation begins. It helps to know how many program lines there are, so you can make room for the table of line addresses. It's also handy to know the size of all the arrays, so you can allocate fixed space for them. ZIP works out both these figures during Pass 1, and builds two fixed-size entries at either end of the 'Compiler Dynamic Area' in the *ZIP Compiler Memory Map Box*.

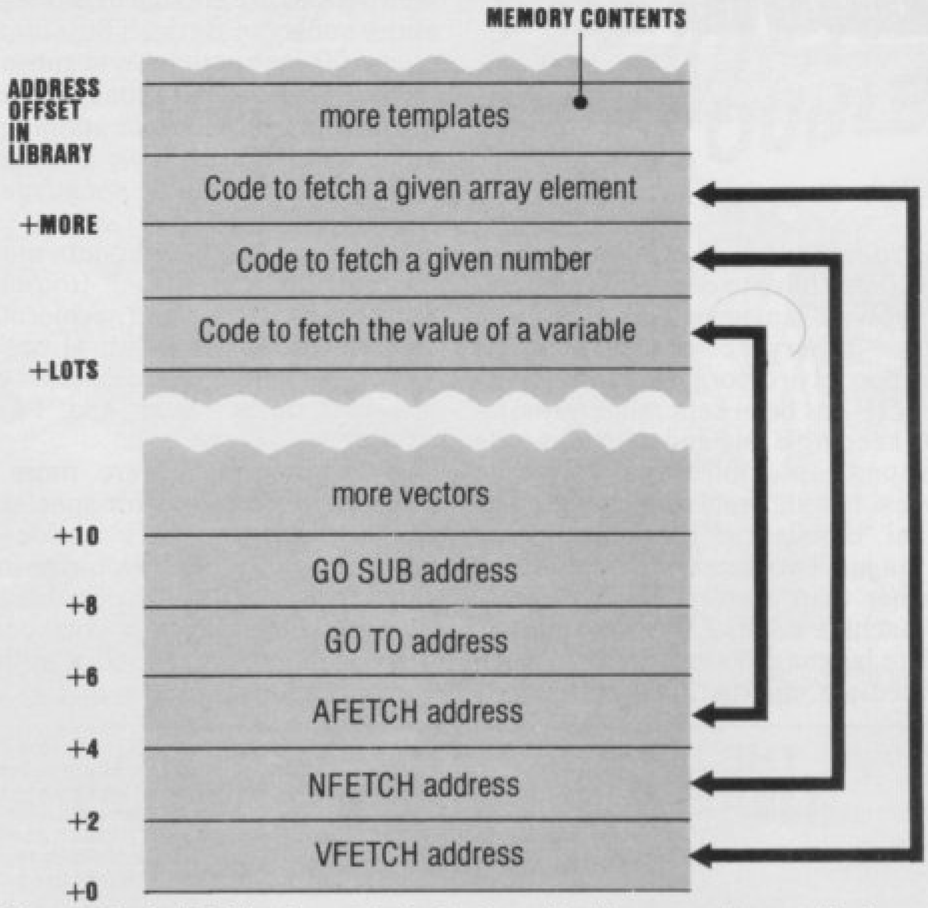
The snag of using a fixed array area is that you can't allow statements like 'DIM A(X+1)' — the size of each array must be known when the program is compiled. We decided that the increased speed of the resultant code made the limitation worthwhile — see Benchmark 7 in the *Benchmark Table*.

ALGORITHMS AND OTHER DISEASES

Programming guru Niklaus Wirth (inventor of the Pascal and Modula languages) coined the phrase 'algorithms + data structures = programs'; 'algorithm' is a posh programming word for 'methods'. In general terms, we've explained how ZIP handles data — now we must look at the algorithms used to convert Basic into machine code.

Compiling a program involves three main steps — reading the original, translating it and storing the results. ZIP shares these out as follows: Pass 1 and Pass 2 read the original. Pass 2 does the translation and stores most of the results. Pass 3 just finishes off storing the compiled code.

LIBRARY 'VECTOR' TABLE



The Library 'Vector' Table above illustrates how the library can be modified without altering the Basic. Using simple offsets into the list (ie. '4' for GO TO, '5' for GO SUB, etc) code is copied from the address indicated in the list to the required template.

ADDING ZIP!

Micro compilers are often messy programs. They have to convert a rather *ad hoc* 'evolved' language — Basic — into what is — by larger computer standards — a horribly disorganised machine code. ZIP has been kept relatively simple by keeping Basic and machine code as far apart as possible. Totally separate routines handle each language. The internal 'translation' routine links the two, via just two numeric variables.

Rather than convert Basic directly into machine code, ZIP uses an intermediate language to link the two. Each intermediate instruction corresponds directly to a short group of machine code instructions, yet less than 70 instructions are enough to represent the entire subset of Basic. The Z80 allows over 600 instructions, yet those don't include simple operations like reading the keyboard, multiplication or division. The listings shows a program in Basic, intermediate language and machine code.

Seasoned machine-coders may criticise the code produced. It could certainly be shortened and accelerated if it was re-written by hand. However, it works just as the Basic does — except it's 150 times faster and 14 bytes shorter!

If the compiler were more complicated it could look for special cases and produce more efficient code — but, frankly, who cares? It would be impossible to generate the best possible code in every instance, even if your compiler ran for months and used a million K. For most purposes a speed-up of 30-100 times, using about as much memory, is ample improvement.

HOW IT WORKS

Once you've decided to use an intermediate code in a compiler, there are three different ways you can process it. The easiest is to write a simple interpreter which scans the intermediate codes one by one and performs appropriate actions. This is the approach used in the UCSD p-System, where the 'p' stands for pseudocode. It produces concise programs but they are slow since they still have to interpreted, although the p-code is closer to machine code than the original language. The simple interpreter must be loaded or the code can't run.

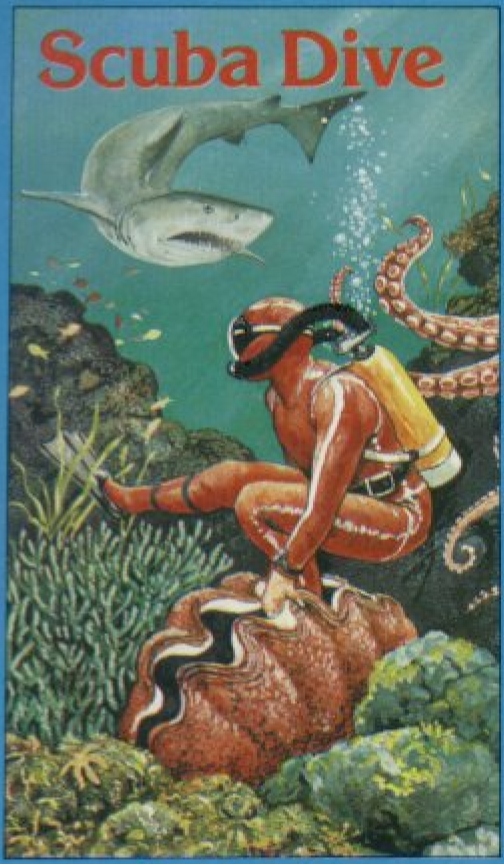
Another technique is to write subroutine calls for every operation — the intermediate code is converted into a mixture of data and GO SUB operations to perform each action. This is 92▶

THE ZIP LIBRARY

First, enter the Basic listing, the *ZIP Library Hex Loader*, and RUN it. On-screen you will be greeted with a message telling you which line address you are working on and another asking for the first byte (in this case, byte 0) of data (ie. you should enter '6E'). Once you have filled all eight bytes, you will be asked to enter the checksum for that line (which for the first line of data is '43'). At this stage, if all is hunky dory, you will pass on to the next line — if you get it wrong, not to worry, you'll get a 'CHECK ERROR' message and you can start over.

```
300 REM ZIP LIBRARY HEX LOADER
305 POKE 23658,8
310 FOR i=53247 TO 54466 STEP 8
315 CLS
320 LET check=0
330 PRINT "LINE ADDRESS ";i
340 FOR j=0 TO 7
350 INPUT "Enter byte ";(j);" ";d$
360 IF LEN d$<>2 THEN GO TO 350
370 LET d=FN d(d$(1 TO 1))*16+FN d(d$(2
TO 2))
380 IF d<0 OR d>256 THEN GO TO 350
382 PRINT AT 2,j*3;d$
385 LET check=check+d
387 POKE i+j,d
390 NEXT j
400 INPUT "Enter checksum ";d$
410 IF LEN d$<>2 THEN GO TO 400
420 LET c=FN d(d$(1 TO 1))*16+FN d(d$(2
TO 2))
430 IF c<0 OR c>256 THEN GO TO 400
440 IF c<>check-INT (check/256)*256 THE
N PRINT "CHECK ERROR: Please retype lin
e": GO TO 320
450 NEXT i
460 PRINT "Please prepare to save the c
ode."
470 SAVE "ZIP LIB 1"CODE 53247,1300
480 GO TO 460
500 DEF FN d(t$)=CODE (t$)-7*(t$>="A")-
48
```

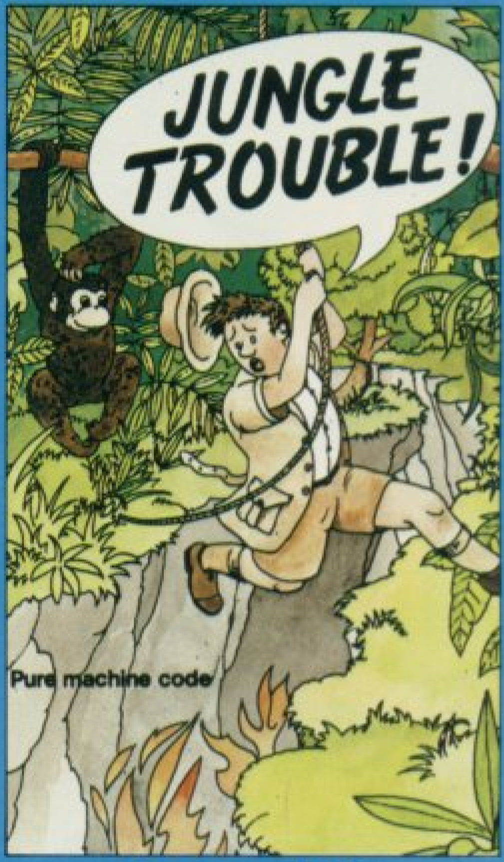
```
53247:6E D4 D7 D0 DB D0 DF D0=43
53255:EB D0 EE D0 F1 D0 F2 D0=FC
53263:04 D1 22 D1 26 D1 2E D1=BE
53271:32 D1 36 D1 42 D1 46 D1=34
53279:4C D1 50 D1 57 D1 5D D1=94
53287:62 D1 66 D1 6A D1 6E D1=E4
53295:72 D1 83 D1 90 D1 9B D1=64
53303:9F D1 A8 D1 A8 D1 A8 D1=DB
53311:A8 D1 A8 D1 9F D1 A8 D1=DB
53319:AE D1 B3 D1 BA D1 C0 D1=1F
53327:D3 D1 E1 D1 EB D1 F9 D1=DC
53335:05 D2 11 D2 1F D2 2A D2=A7
53343:2D D2 33 D2 36 D2 3C D2=1A
53351:48 D2 4D D2 4F D2 58 D2=84
53359:5E D2 67 D2 6B D2 6F D2=E7
53367:4D D2 6F D2 16 D3 2D D3=49
53375:44 D3 62 D3 86 D3 95 D3=0D
53383:B5 D3 CD D3 12 D4 32 D4=14
53391:01 2D D2 01 33 D2 01 36=3D
53399:D2 01 3C D2 01 48 D2 01=FD
53407:4D D2 01 4F D2 01 58 D2=6C
53415:01 5E D2 01 67 D2 01 6B=D7
53423:D2 01 6F D2 01 4D D2 01=35
53431:6F D2 01 16 D3 01 2D D3=2C
53439:01 44 D3 01 62 D3 01 86=D5
53447:D3 01 95 D3 01 B5 D3 01=C6
53455:CD D3 01 12 D4 01 32 D4=BE
53463:2A 00 00 E5 21 00 00 E5=15
53471:E1 11 00 00 2B 29 19 5E=BD
53479:23 56 EB E5 C3 00 00 CD=D9
53487:00 00 C9 E1 22 00 00 E1=AD
53495:22 00 00 E1 22 00 00 21=46
53503:00 00 22 00 00 2A 00 00=4C
53511:ED 4B 00 00 09 22 00 00=63
53519:ED 5B 00 00 B7 ED 52 28=66
53527:06 7C AB E6 80 28 04 2A=E6
53535:00 00 E9 CD 6F D2 E5 3E=1A
53543:02 CD 01 16 CD 6B 0D E1=0C
53551:22 00 00 E1 D1 EB 73 E1=13
53559:D1 EB 01 00 00 2B 29 09=1A
53567:73 23 72 E1 C1 ED 69 E1=E1
53575:D1 CD 12 D4 E5 E1 D1 19=34
53583:E5 E1 D1 EB B7 ED 52 E5=5D
53591:E1 D1 CD CD D3 E5 E1 7D=62
53599:CD 9B 22 E1 CD 44 D3 E1=30
53607:CD 62 D3 E1 CD 16 D3 E1=7A
53615:CD 2D D3 E1 7D B7 28 02=0C
53623:2E 08 3A 91 5C E6 F7 00=3A
53631:85 32 91 5C E1 3A 91 5C=AC
53639:E6 FD 85 85 00 00 32 91=B0
53647:5C 11 00 00 01 00 00 CD=3B
53655:3C 20 EB E9 E1 CD 32 D4=E4
53663:E1 3E 16 D7 D1 7B D7 7D=AC
53671:D7 E1 3E 17 D7 7D D7 E1=19
53679:6E 26 00 E5 E1 11 00 00=6B
53687:D5 E9 C5 C1 ED 68 26 00=BF
53695:E5 E1 7C B5 28 0D 7C E6=8E
53703:80 20 05 21 01 00 18 03=E2
53711:21 FF FF E5 E1 7C E6 80=C7
53719:28 07 7C 2F 67 7D 2F 6F=5C
53727:23 E5 E1 7C B5 21 00 00=3B
53735:20 01 23 E5 E1 D1 CD B2=5A
53743:D4 21 00 00 38 03 28 01=59
53751:23 E5 E1 D1 CD B2 D4 21=2E
53759:00 00 38 01 23 E5 E1 D1=F3
53767:CD B2 D4 21 00 00 30 01=A5
53775:23 E5 E1 D1 CD B2 D4 21=2E
53783:01 00 38 03 28 01 2B E5=75
53791:E1 D1 B7 ED 52 28 03 21=F4
53799:01 00 E5 E1 7D D7 E1 7C=78
53807:B5 CA 00 00 CD 95 D3 E1=95
53815:D1 CD 86 D3 E5 E1 D1 B7=45
53823:ED 52 21 01 00 28 01 2B=B5
53831:E5 3E 02 CD 01 16 CF 08=E0
53839:E1 7C B5 28 04 21 01 00=60
53847:E3 E1 7C B5 20 01 E3 E1=DA
53855:7D 2F 6F 7C 2F 67 23 E5=35
53863:E1 22 76 5C C1 CD 3D 1F=BF
53871:AF CD 01 16 CD 6E 0D 21=FC
53879:00 5B 0E 00 AF 32 08 5C=AE
53887:3E 8F D7 3E 08 D7 3A 08=03
53895:5C FE 0D 28 38 FE 0C 28=F9
53903:22 FE 30 38 10 FE 3A 30=00
53911:0C 47 3E 06 BD 28 DD 78=D1
53919:77 2C D7 18 D7 FE 2D 20=B4
53927:D3 7D B7 20 CF 0E FF 3E=41
53935:2D 18 ED 7D B7 28 C5 3E=91
53943:20 D7 3E 08 D7 3E 08 D7=31
53951:2D 28 B4 18 B7 77 11 00=60
53959:00 21 00 5B 79 B7 28 01=D5
53967:23 7E D6 30 30 0A AF B9=49
53975:28 01 3C BD 28 9E 18 2A=2A
53983:E5 EB 11 CC 0C B7 ED 52=AF
53991:38 10 20 04 FE 08 38 0A=B4
53999:EB E1 23 7E FE 0D 20 FA=92
54007:18 82 19 29 E5 29 29 D1=E4
54015:19 EB E1 83 5F 30 01 14=0C
54023:18 C6 79 B7 28 07 7A 2F=E6
54031:57 7B 2F 5F 13 EB C9 7D=A4
54039:FE 08 28 0D B7 20 05 FD=14
54047:CB 53 BE C9 FD CB 53 FE=BE
54055:C9 FD CB 54 FE C9 7D FE=27
54063:08 28 0D B7 20 05 FD CB=E1
54071:53 B6 C9 FD CB 53 F6 C9=AC
54079:FD CB 54 F6 C9 7D FE 08=5E
54087:28 0C 30 12 3E F8 FD A6=4F
54095:53 85 32 8D 5C C9 3A 8E=84
54103:5C F6 07 32 8E 5C FD CB=3D
54111:57 EE C9 7D FE 08 28 11=CA
54119:30 18 3E C7 FD A6 53 67=AA
54127:7D 07 07 07 B4 32 8D 5C=61
54135:C9 3A 8E 5C F6 38 32 8E=DB
54143:5C C9 FD CB 57 FE C9 EB=F6
54151:29 29 29 29 29 19 11 00=F7
54159:58 19 6E 26 00 C9 21 00=EF
54167:D6 E5 75 54 5D 13 01 CF=C4
54175:00 ED B0 01 DB 1D 11 07=AB
54183:00 E1 19 2B 3E 1A 71 23=11
54191:70 19 3D 20 F9 C9 21 38=01
54199:1F E5 3E CF ED 47 ED 5E=90
54207:18 0C 7F 31 39 38 34 53=CC
54215:4E 47 26 4A 41 53 7C B5=CA
54223:20 02 CF 05 0E 00 CB 7C=4B
54231:28 08 7C 2F 67 7D 2F 6F=5D
54239:23 0C CB 7A 28 08 7A 2F=4D
54247:57 7B 2F 5F 13 0C C5 EB=2F
54255:7C 4D 21 00 00 06 10 CB=CB
54263:21 17 ED 6A ED 52 30 03=01
54271:19 18 01 0C 10 F1 67 69=0F
54279:C1 0D 20 06 2F 67 7D 2F=36
54287:6F 23 C9 7C B7 20 0C 7D=37
54295:6C 06 08 29 17 30 01 19=04
54303:10 F9 C9 4D 06 10 21 00=56
54311:00 29 CB 11 17 30 01 19=66
54319:10 F7 C9 CB 7C 28 0A 7C=C5
54327:2F 67 7D 2F 6F 23 3E 2D=3F
54335:D7 7C 4D 21 01 00 06 04=CC
54343:E5 29 54 5D 29 29 19 10=3A
54351:F7 E5 67 69 AF D1 ED 52=6B
54359:38 03 3C 18 F9 19 4F B0=A0
54367:79 20 04 CB 43 28 04 C6=9D
54375:30 D7 04 1D 20 E6 C9 F5=EC
54383:3E FE DB FE 1F 38 07 3E=B1
54391:BF DB FE 1F 30 04 F1 FF=DB
54399:ED 4D 2A B2 5C 2B F9 2B=C1
54407:2B 22 3D 5C AF 32 71 5C=94
54415:3D 32 70 5C 3C CD 01 16=5B
54423:CD 6E 0D 21 3B 5C CB 9E=69
54431:23 CB EE 3E 15 11 91 13=E4
54439:CD 0A 0C FB C3 A9 12 00=5C
54447:00 00 00 B7 CB 7C 28 07=2D
54455:CB 7A 20 07 AF 37 C9 CB=E6
54463:7A 20 02 ED 52 C9 00 00=A4
```



Spectrum 48k
Oric 16-48k
Commodore 64



Spectrum 16-48k
Oric 16-48k
Commodore 64, Atari



Spectrum 16-48k



Oric 48k



Oric 16-48k

DURELLsoftware

available from

W. H. SMITH · LASKYS · GREENS at DEBENHAMS BOOTS · COMPUTERS FOR ALL · MARTINS · SPECTRUM CENTRES · and many other retail outlets.

or order direct from
CASTLE LODGE, CASTLE GREEN, TAUNTON, SOM. TA1 4AB

MAIL ORDER

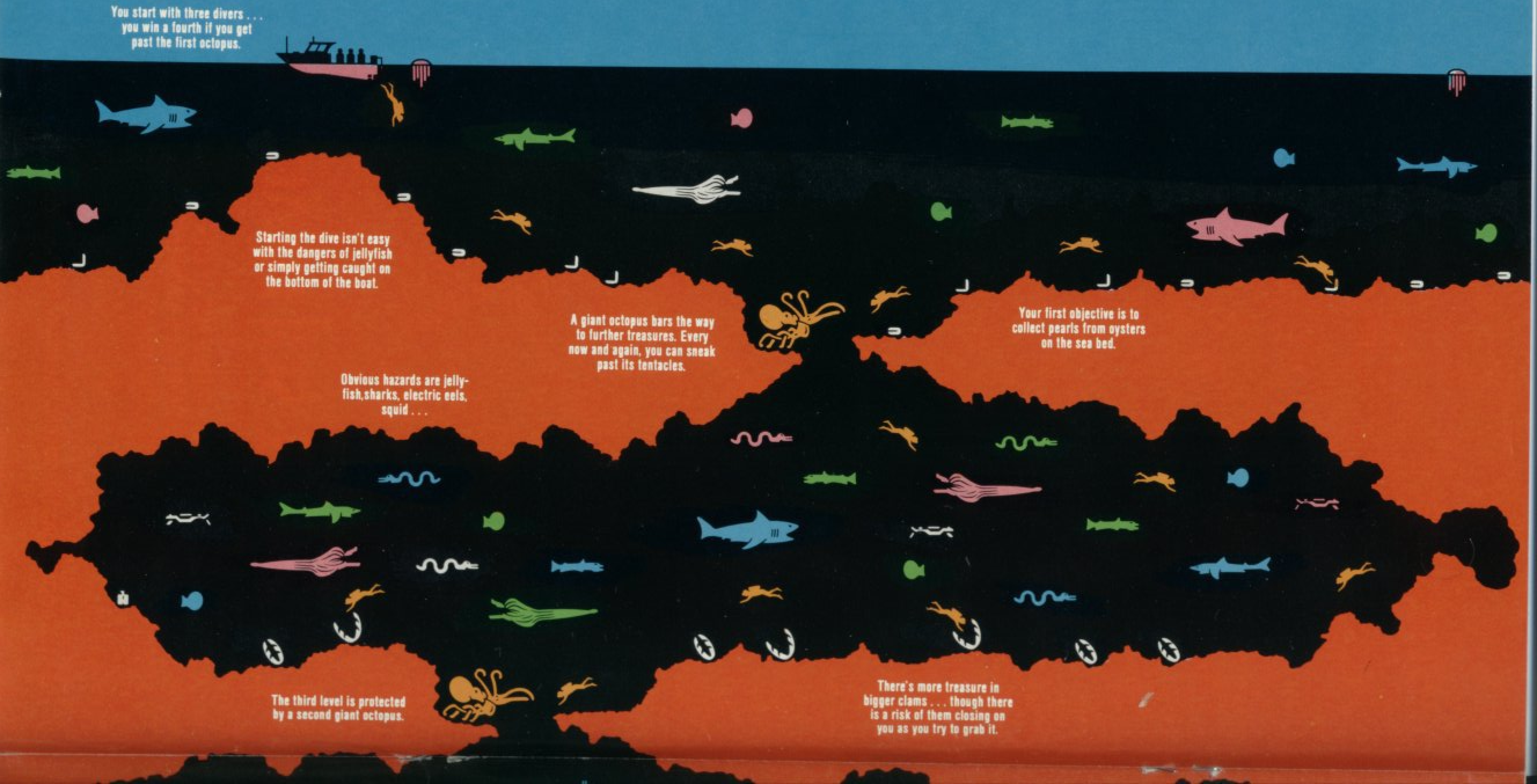
SPECTRUM software £5.50 all other software £6.95
Price includes postage and packing

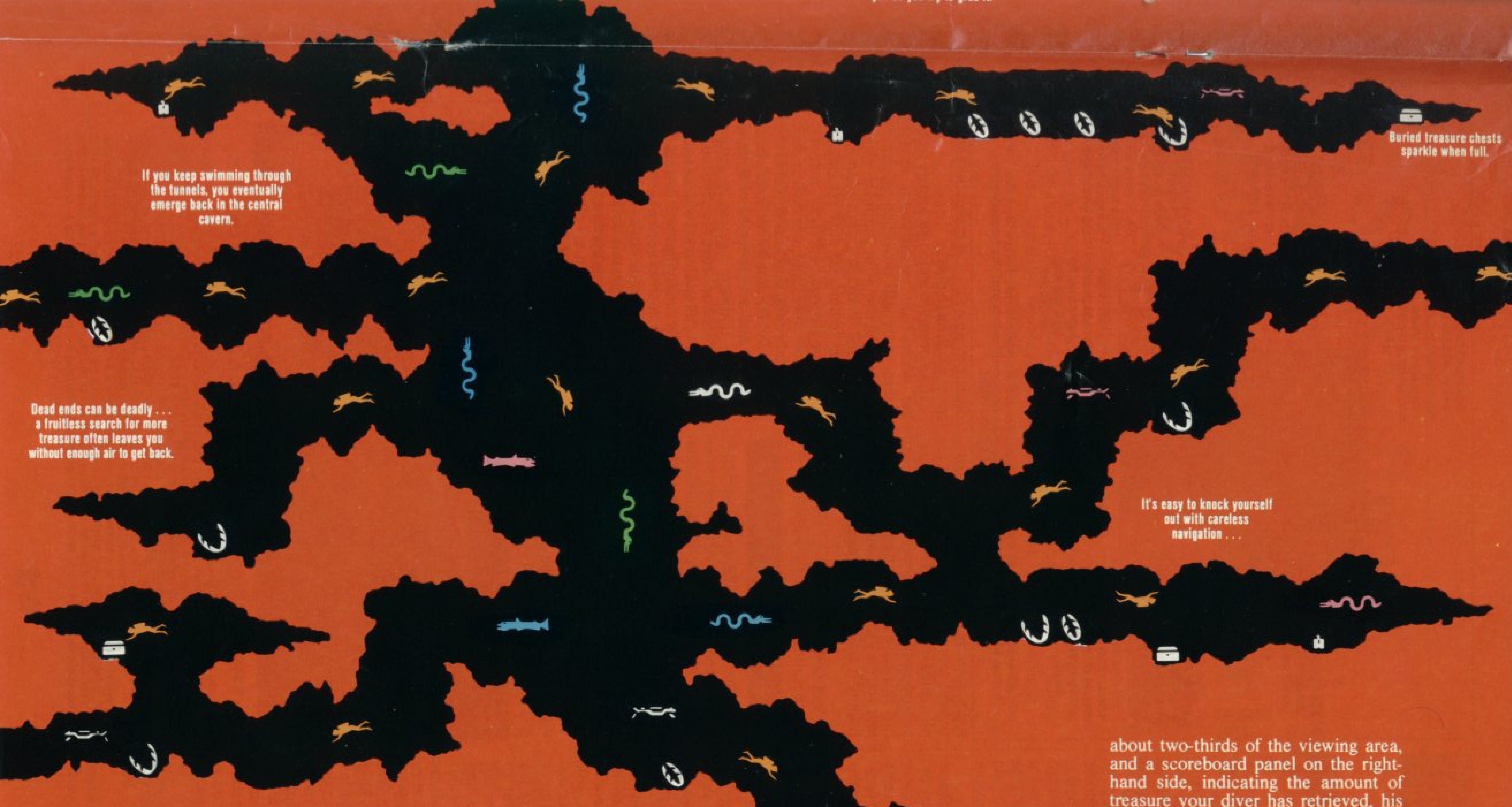
TYPE OF COMPUTER: ....
CASSETTE TITLES:
1) ....
2) ....
3) ....
NAME: ....
ADDRESS: ....
....
....
....

Overseas customers please allow £1.00 extra

WET SCREAMS

Into the jaws of death dives* Mike Mepham *in search of treasure. Plumb the depths of Durrell's underwater nightmare and discover why Scuba Dive is Mike's fave Speccy challenge of the month.





Even before starting the game proper, marvel at the coding that has gone into the animation of *Scuba Dive* as you are given a fish's eye view of the marine environment; it's rather like looking into a tropical fish tank. The object of the game is to manipulate your scuba diver around the inky black depths, collecting pearls and sunken treasure and taking them safely to your cabin cruiser on the surface. The trouble is, of course, there are many hazards for your minuscule marine explorer to avoid — like sharks, shoals of fish, squid and giant octopuses — all of which can kill! He's also likely to knock himself unconscious on the sea bed or run out of oxygen.

In games like this, with all the risks to life and limb, one would expect the diver to be zapping and blasting sharks and squid all over the screen with a harpoon gun and an inexhaustible supply of ammunition. In *Scuba* the diver is totally defenceless and always vulnerable to attack — anywhere, anytime.

Your screen is split into two parts — the square playing section that takes up about two-thirds of the viewing area, and a scoreboard panel on the right-hand side, indicating the amount of treasure your diver has retrieved, his depth, how much air he has left and the value of treasure he is holding.

Pearls are contained in oysters which are constantly opening and closing — giant pearls are in giant clams. Treasure lies in chests that sparkle when full, less so when empty.

The picture you see is constantly changing and as the diver approaches the edge of the screen, the scene is shifted, jumping sideways on the screen, but not moving the diver in relation to his surroundings. In this way, a seemingly continuous sea bed has been

MARK WATKINSON

created. There is a 'bottom', but supply of oxygen seems the only limitation to how far you can swim either way.

PLAYING THE GAME

Author Mike Richardson has done games players proud with *Scuba*. The game was tested on a 48K Spectrum using the keyboard controls; if you want to use a joystick, then the keys can be programmed to suit. And in addition to the major plus point of offering programmable control keys is the facility of being able to SAVE and LOAD score tables. *All* publishers please note!

Four skill levels are available, '1' being the easiest and '4' the most difficult. At level '4' the sea is thick with sharks, squid and all the rest. Three divers are available for play at the start, but a bonus diver joins the party on board the boat if one of the others manages to gain access to the caves containing treasure chests.

Play is started by touching the Space key. As the diver enters the water, the first problem is to get him safely away from the hull, because if there aren't lethal jellyfish floating on the surface, then he can always kill himself on the bottom of the boat.

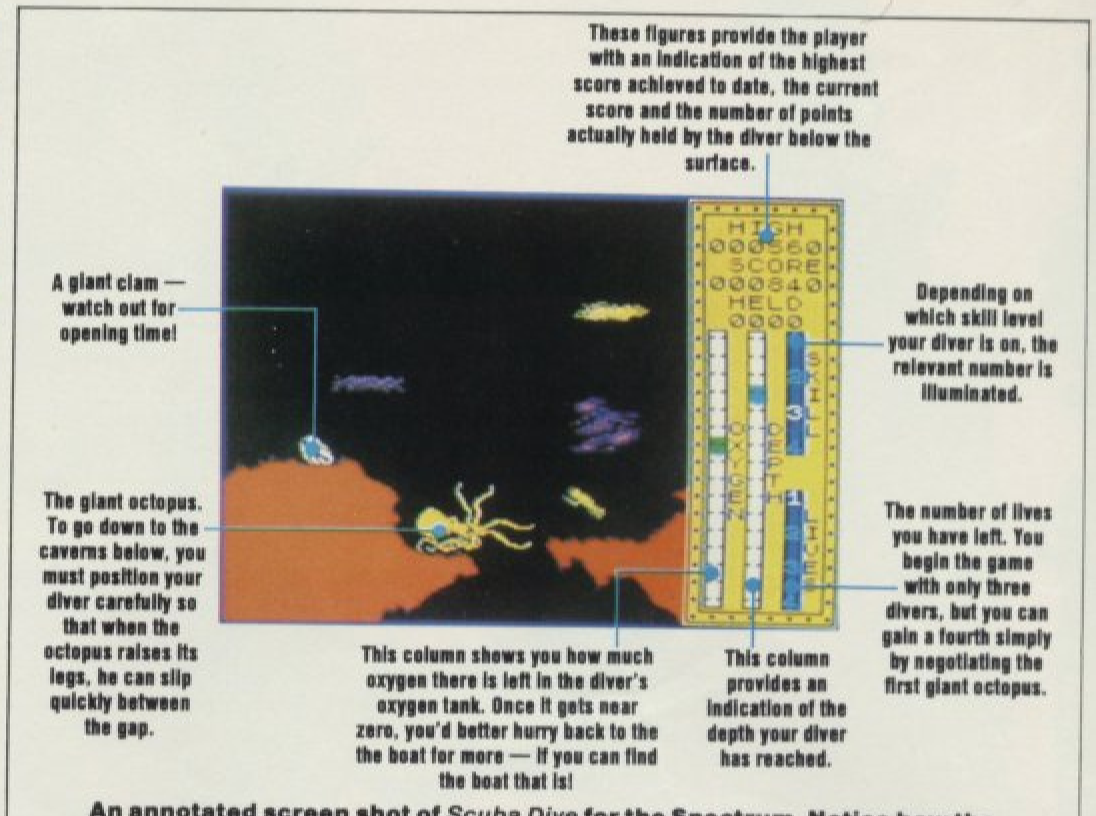
Collecting oyster pearls is a matter of waiting until the oyster opens and then touching it with the diver; there's a maximum value of pearls and treasure he can carry, which varies according to the skill level of play. But whatever the value, it's not added to the score until the goodies are stashed safely back on the boat. This is probably the most frustrating part of the game. You could be holding the maximum value allowed and just by touching a rock you've knocked your diver unconscious and everything's gone.

Once you've mastered the rotate controls (a la *Asteroids*), the first real challenge is to negotiate the tentacles of the first octopus as it stands guard over an undersea cave system. The caves are home to dangerous sea creatures and giant clams containing yet more valuable pearls. But beware . . . the clams can knock a diver unconscious if you're not careful. And another giant octopus guards the entrance to yet another cave system — one where the diver can help himself to treasure from glowing chests.

There are three distinct levels to the game, each growing progressively more difficult. The first involves gathering oyster pearls and returning them to the boat, and for the second, you must negotiate an octopus and gather up clam pearls (then re-negotiate the octopus and return to the boat); finally, having done all that, it's time to get into the lower cave complex. Anyone making it that far deserves that the surprise awaiting them stays as a surprise — sorry, but my lips are sealed!

Naturally while all these activities are going on, the diver's oxygen is gradually running out, although every successful return to the boat will have replenished his supply. That sounds easy enough until you discover that the boat has drifted from its original position, and you may just have some difficulty in finding it again on returning to the surface.

And that's the game . . . frustrating, exasperating and totally addictive.



An annotated screen shot of *Scuba Dive* for the Spectrum. Notice how the screen is divided into the actual playing area on the left and the scoreboard panel on the right.

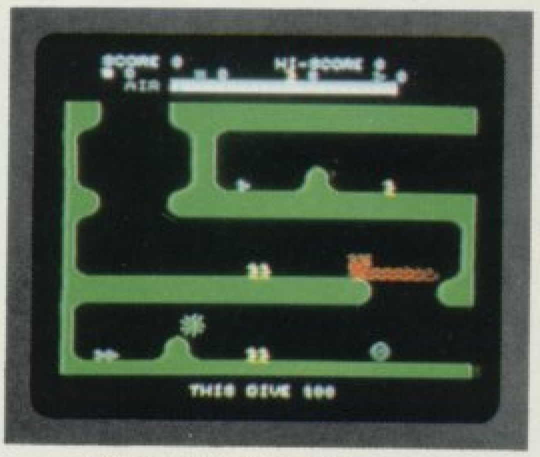
CONCLUSION

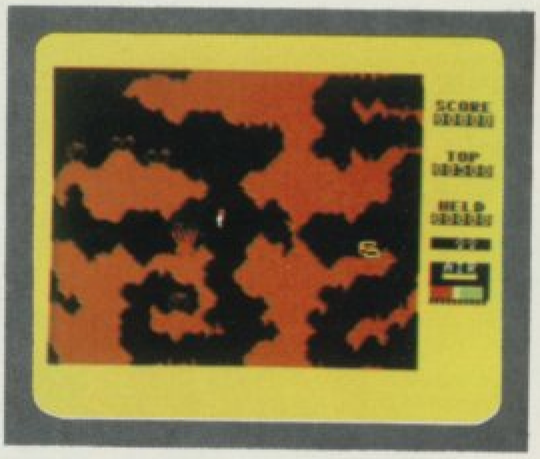
No doubt some genius (perhaps Mike Richardson himself) will bring even greater realism to computer animation in the future, for the *Scuba Dive* experience is somewhat akin to the enjoyment people had when the first *Mickey Mouse* cartoons came flickering onto the silver screen.

Managing Director of Durell, Robert White, says that many hours of hard discussion went into the decision not to give the diver any form of weapon; that decision was right . . . it makes the game *more* exciting to play.

It's a program which gets away from the wham-bam, mainstream arcade games while still providing an entertaining scenario with graphics which will not pall. Perhaps it's not a game for the seven- to ten-year olds, although it does have periods of frenetic activity. But on the other hand, nor does it need great intellectual skill to understand and operate; *Scuba* simply requires sharp reactions and a high degree of manual dexterity. Ultimate success in getting your diver back on board the boat, totally against the odds and loaded with treasure, is immensely satisfying and it can easily take an hour or so to accomplish.

And more than anything, Mike Richardson and Durell Software have provided another important target for the software writers to aim for and improve on. For that reason alone it's worth buying it to see what all the fuss is about. YS





THE 'WE DON'T WANT TO GLOAT, BUT . . .' DEPT.

As *Scuba Dive* is also available for the Oric and Commodore 64 micros, we thought you might be interested to see how the versions compare. Both are a bit of a disappointment after the Spectrum original but, to be fair, the program for the Oric has been cut down substantially to run within 16K — and as such is a good conversion. Both versions feature simpler displays utilising unexceptional graphics and animation. There also seems to be a lot less going on on-screen. The diver moves in just four directions (up, down, left and right) rather than the subtle turning and acceleration exhibited on the Spectrum version.

The caves on the Oric conversion are limited to a simple maze (harking back to the ZX81 graphical days), inhabited by comic-looking octopusses and one creature with the amazing ability to appear and disappear at random. But worst of the bunch is the Commodore 64 version, with even the giant octopus being replaced by a trapdoor.

Rumours of new launches abound on the games software scene — and there's no love lost between competing software houses, eager to market an original product. Get the up-to-the-minute report from* Ron Smith, *whose beady eyes remain fixed on the software horizons.

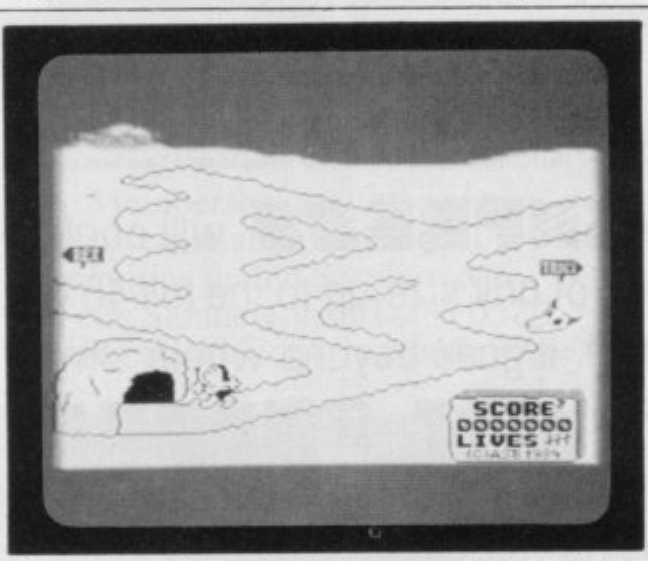
RUMBLES

From the maker's point of view, there's often a lot to be said for announcing new products some time before they're due to be launched; it helps get the 'buzz' going around. But this can sometimes be taken a little too far, a trap that some believe Melbourne House fell into back in the autumn of last year.

At the launch of *Terror-Daktil 4D* last October, the over-excited Melbourne House management proved quite unable to contain itself and decided to tell the world about its next great new game — *Sherlock Holmes*. Since then, having whetted our appetites with claims that this would be the hottest property since *The Hobbit*, everything's gone quiet. So, *YS* hit the trail to find out what's happening.

We spoke to MH publicity manager, Paula Byrne: "It should be available some time in May", she hazarded, stressing that people need not worry because "we never cash cheques until the order is about to be despatched". In fact, "because *Hurg* was late, this made us wary about announcing firm release dates". Often, it seems the problems really lie with the programmers. They become over-optimistic and give unrealistic completion dates, and that's when it all starts to hit the fan. Confirming this, Ms Byrne said "we just let the programmers get on with it". Of course, the upshot of working this way is that it gives them the extra time necessary to come up with a truly professional and highly polished program.

Details are still fairly sketchy, but we're told it's an adventure game which includes both graphics and text, and one that's been developed from *The Hobbit*. If that's true, then at a cost of £14.95, MH might well have another winner on its hands.



A sneak preview of the prehistoric pandemonium in Ugh! — one of the new games to be launched from Softek.

SCARFACE MEETS SPECTRUM

Another new game, and the latest from the MH stables, is a combined adventure/strategy game called *Mugsy*. It has an assortment of gangsters with their respective molls, and all these hoods have to be 'talked' to in their own vernacular or they refuse to understand your commands.

Mugsy, priced at £6.95, includes some "very good graphics", claims Ms Byrne; "you can even see the expressions on the characters' faces". Sound's great . . . I can hardly wait!

MORE HYPE FROM VIRGIN

New software houses seem to spring up overnight, and one that 'sprang' over the course of last year — albeit a little shakily to begin with — was Virgin Games.

The company's just announced a new game called *Hype*, and strange to say it's not a computer game but, in fact, an old fashioned, common or garden board game. The idea is to get into the record business and promote your band (something which Virgin ought to know a thing or two about). But why, we asked Virgin's managing director, Nick Alexander, the move towards more traditional games? Was Virgin going to stop producing computer games?

Denying rumours that he was also going into the business of manufacturing fluffy toys, he said "It was just a series of coincidences. Since we started to produce computer games, we have constantly been on the look-out for new ideas. And one source of these ideas came from people who had designed board games that had never actually been marketed. So we looked at these suggestions, and found that while some of them could be successfully transferred to computer, others were better off left in their original form".

However, decision taken, the reality of it apparently proved rather more of a problem. Says Nick, "The whole thing was much more difficult than we first thought. For example, *Hype* should have been in the shops before Christmas, but due to immense production problems, we found we couldn't get it out until now". He went on, "board games certainly have more value, socially, than computer games". And to back up this conviction, he plans to continue producing them throughout the year. At the same time he assures all Virgin computer games enthusiasts that efforts in producing new software will be maintained.

BACKING OCEAN'S HUNCH

After the success of its *Hunchback* program, Ocean has decided to produce a sequel which is to be called *Quasimodo's Revenge*, and it will be launched at the *Your Computer* fair later this year. Ocean's managing director, David Ward, says "the new program will have a dozen different screens — and will be the most fantastic thing of all time". Which may even be true, especially if it's anything like the original.

Ocean is continuing in its prolific way with at least four other new titles in the pipeline, though some are as far away as six months from general release. The first, to look for is *Cavalon*, a true arcade game in the sense that it has been licensed from the arcade original. It's based on the story of the court of King Arthur, where the player has to rescue the distressed damsel, and at the same time pick up the eight pieces of a door. There are knights trying to hinder your progress, but help is at hand in the form of Arthur's trusty sword, Excalibur. Should you manage to pick this up, you'll be protected; for about 20 seconds!

The second title is *Gilligan's Gold*. Here you play the part of Gilligan and the idea is to collect sacks of the yellow stuff; you are equipped with a wheelbarrow to speed things up a little. It also features a scrolling maze, and plays the old Laurel and Hardy song 'Trail of the Lonesome Pine'.

Details on two other games were less forthcoming, but one is to be called *Chinese Juggler*. It was concocted, unlikely as it seems, in Hungary and Ocean is even reported to have sent some machines out there to get the program written. The finished product, according to David Ward, "offers some truly outstanding graphics". And in hushed tones added, "it's all very top secret at the moment". But if you want a sneak preview, try taking a look at the version that's been around for ages for the CBM 64 . . .

And finally, one more for the pot, a sun-tanned Sandy White has confirmed to us that he expects his *Ant Attack* follow-up to be ready for Christmas. YS

***Your Spectrum* is continuing to scour the country's computer clubs to get the undiluted thoughts, feelings and impressions of dedicated games enthusiasts — thus providing our readers with unbiased evaluations. Any club wishing to offer their reviewing services should contact Ron Smith, Spectrum Soft, 14 Rathbone Place, London W1P 1DE.**

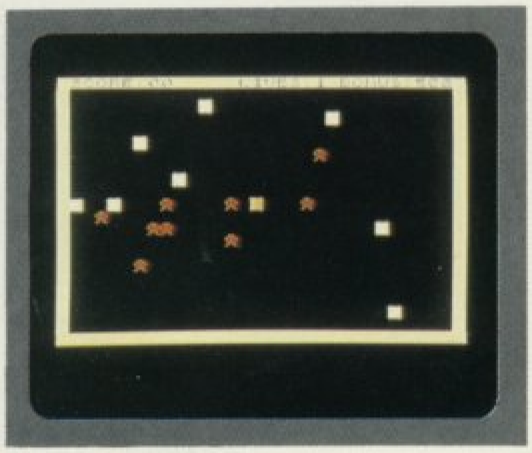
This month we visited the Heald Green Computer Club (which meets fortnightly at the Public Hall, Outward Road, Heald Green, Cheadle, Cheshire, on Wednesdays from 7.30 to 10.30pm). Three of the club's leading members each played the games we gave them for review and in some cases found them so well-conceived and written, went completely over the top with the superlatives. In others, the games were deemed to be re-hashes of old ideas and evaluated accordingly.

The club was formed four months ago, and has already managed to attract over 40 enthusiasts. All members are being brought up to the same standard of proficiency by the more experienced members who are running formal courses in both Basic and machine code.

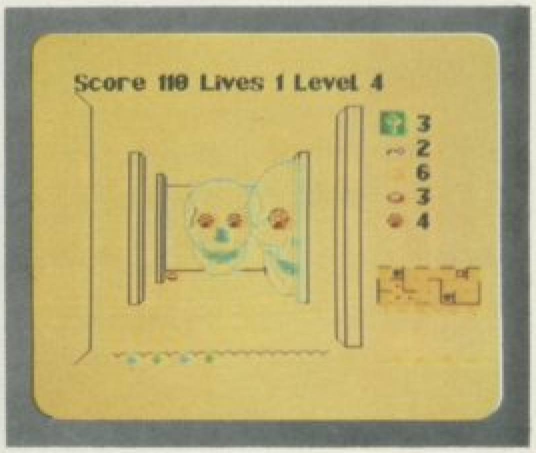


Heroes of the Heald Green Computer Club — Jon Hall, Tony Samuels and Mark Knight.

Membership is offered at a very reasonable rate, and all enquiries should be referred to Tony Samuels on 061-437 0580 (after 6pm). Tony, incidentally, is currently working on a new program for the Spectrum (called *Ugh* — see *Rumbles* in this issue) which will be available from Softek in the near future.







SPECTRON
Virgin Games/£5.95

Control the 'Spectron fighting robot' and avoid the various obstacles while blasting the meanie robots to bits; and all this takes place in an area that's contained by an electrified fence.

Mark This variation of *Berserk* is fairly fast, but doesn't seem to live up to its boast of having been written in 100 per cent machine code. And it doesn't make much use of colour or the Spectrum's graphics capabilities. 4/10

John This program's best feature is the instructions which appear on-screen at the start. These explain clearly how to play and, should the user forget, there's always the cassette insert. 4/10

Tony The idea behind this game is a variation on a simple theme, but is made more complex by having several different enemies. It's a pity only simple block graphics are used. 5/10

ROBOT RIOT
Silversoft/£5.95

Rioting robots run amok in this maze game, and your job is to stop them as quickly as possible by carefully controlling your robot to plant bombs that will destroy the mutinous menaces.

Tony This is essentially a *Pacman* game in reverse. Instead of eating dots, the player has to lay them. The theme is original and highly addictive, but in play, feels a little too slow for comfort, and the keys aren't too responsive. 7/10

Mark The player's robot is very controllable, and the first level starts at a reasonable speed — which increases with successive levels. With superb graphics and colour, this must surely be the best *Pacman* game ever. 8/10

John Everything involved in this game is excellent, from the impressive on-screen scoring to the cassette inlay. This is undoubtedly Silversoft's best. 9/10

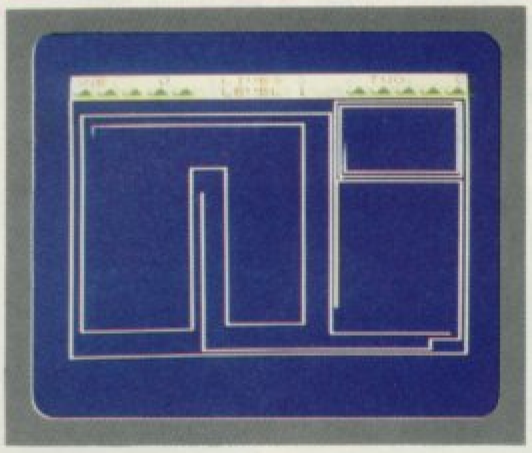
SKULL
Games Machine/£6.95

This is a 3D maze game in which the player has to collect coins, crosses, gemstones and keys that'll open gates. All the while one has to avoid the skulls that stay in constant pursuit.

Tony 3D maze games may not be exactly new on the Spectrum, but this idea is — and it's also very exciting. *Skull* is not what is usually understood by the word colourful, but its use has produced some quietly stunning graphic effects. 8/10

John The skulls come close to the standard of a professional illustration. The treasure doesn't reach the same standards of excellence, although it can still be identified. 9/10

Mark The instructions on the cassette insert are very comprehensive. There's also an keyboard overlay to help the player get the hang of the controls. Highly recommended. 9/10



SUPER SNAILS
Games Machine/£5.95

Snails are being bred in a laboratory and two of them escape — one of which is controlled by the player. As they move they leave a trail of Super Glue. And it's your task to trap the other snail, at the same time avoiding your own trail and that of the fellow escapee — all in an effort to avoid a sticky end.

Mark The speed of this game, which can be selected by the player, ranges from very fast to unplayable. The graphics are almost non-existent . . . even the snail is only one pixel big. 6/10

John Use of colour is fairly dull, with the screen starting off about 80 per cent blue and two white snails leaving two white trails as they go. 5/10

Tony The speed ranges from fast to ridiculous; it's virtually unplayable, and made worse by the two-colour screen and the distinct lack of graphics. 4/10



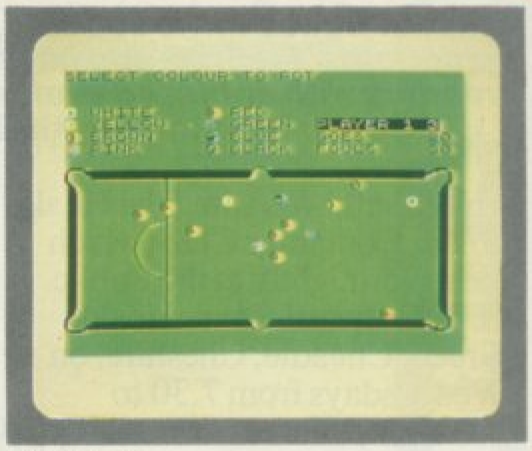
PI-EYED
Automata/£6.00

Automata's *Pi-man* is on the great slippery slope and sees life through a drunken haze. The player's job is to help him negotiate his way across a busy main road and into the nearest pub.

John This is definitely one of the most amusing new games, in concept at least, to come on to the market. Sometimes the action seemed a little too fast, especially as the *Pi-man* became even more inebriated. 8/10

Mark Colour was used well throughout, but not spectacularly; the pub scenes were the best in terms of graphical excellence. But the game opens a little slowly, with the keys not responding too well to the player's directions. 7/10

Tony Ten out of ten for novelty value, but as is so often the case, the novelty soon wears off. There's also an amusing little song on the reverse of the tape. 8/10



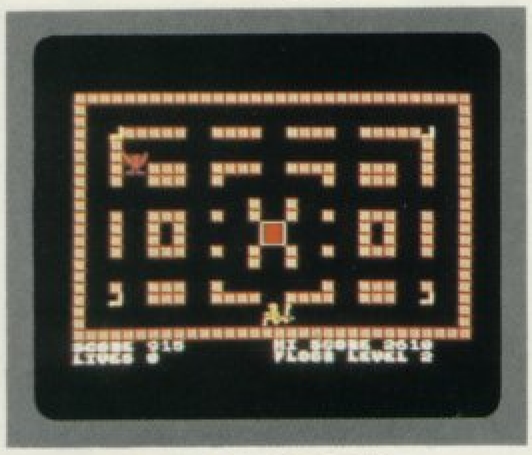
SNOOKER
Visions/£9.95

A computer simulation of the real-life game that theoretically allows the player to enjoy snooker without the need of a cumbersome table. It's a fairly new idea on the Spectrum.

John The speed is very irregular. Still, it's not easy to adequately reproduce the movement of a ball on screen. More effort could have gone into the graphics; the balls are not very well defined. 6/10

Tony Problems occur with colour. For example, it's not possible to have a green ball on a green table. Some of the balls are therefore represented by non-standard colours. 6/10

Mark Comprehensive instructions appear on the cassette insert, as you'd never remember them all straight away. The program has one slight technical problem — if the ZX Interface 1 is connected, it doesn't work. 7/10



HOUSE OF THE LIVING DEAD
Phipps Associates/£5.95

Another maze game where each maze represents (at a guess) a floor in a haunted hotel. The aim is to tear around picking up the four sections of a key, and assembling them while avoiding the evil spirits. This done, the player can move on to a harder level with more sprites.

John This game has a highly original theme but looks rather like *Pacman* without the dots. The graphics are extremely good, with flapping bats and a skeleton with a funny walk. 7/10

Tony Instructions were a little scarce, screen layout is exceptional but there could have been a greater variety of monsters in a 48K program. 8/10

Mark The speed of this game is just about right and colour has been used to good effect, enhancing the well-defined graphics. It also includes a very nice loading display. 8/10



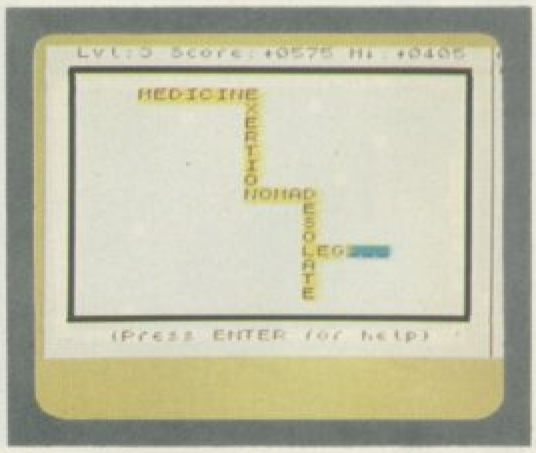
NAANAS
Micro-Gen/£5.95

A pet monkey is hiding up a tree and mischievously trying to drop both bananas and coconuts on to its owner's head. All the player has to do is stand on the ground endeavouring to catch as many bananas as possible, while carefully avoiding the heavy coconuts.

Tony A highly original game, but one that falls short of being addictive because not a lot seems to happen. The other problem is that the choice of colours is not as good as it could be. 5/10

John A fast, uncontrollable game that uses cyan as its main colour — which doesn't help the tedium. The graphics are good, especially when the monkey's owner slips on the bananas. 6/10

Mark It's a pity that the playing speed can't be altered, because the game often seems too fast to control properly. You can, however, define your own keys. 6/10



XANAGRAMS
Postern/£6.95

This looks like some sort of educational cross between Hangman, anagrams and crosswords. The game begins with a line of dashes that show where the letters go, with the letters appearing — jumbled up — on the right-hand side of the screen.

Tony There's minimal use of colour, but that's not surprising in a text-only game. But more use of it might just have encouraged the very young to play the game a little longer. 6/10

John This is a user-friendly program with a well laid-out screen that's easy to see and a useful Help option. An enjoyable game to play. 7/10

Mark This is a fairly interesting program to play — especially so if you like your word games even more difficult, with the clues being given as a jumbled-up mess! Overall, absorbing and good to while away a few hours. 7/10



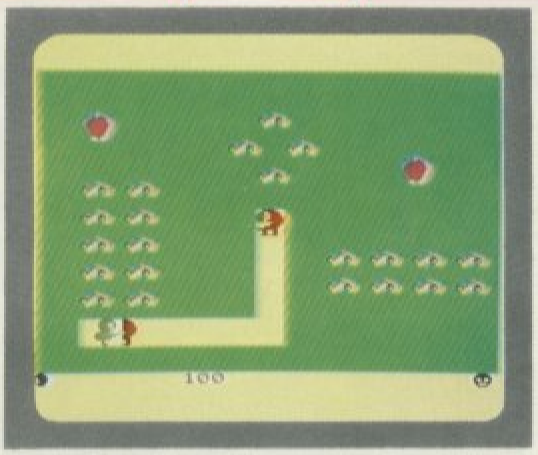
1984
Incentive Software/£5.50

A game of government management that asks the player to use his or her judgement on the complex inter-related issues that govern the performance of the national economy. As the British Prime Minister, all you have to do is stay in power until 1999!

Tony Speed, although not particularly important, could probably have been improved upon; the program looks as though it's been written in Basic. But the real enjoyment comes from the daunting task of running the country. 7/10

John Being more or less a text-only game colour and speed are of little importance; but they're used to good effect in both cases. 8/10

Mark The idea behind this game is both typical and interesting; it's ideal for economists and anyone else who would like to have a go. 7/10



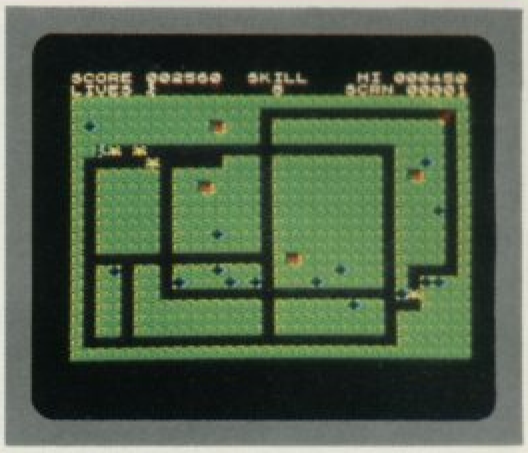
DINKY DIGGER
Postern/£6.95

The player takes on the role of the Dinky Digger, whose job it is to dig his way underground and eat all the cherries. At the same time, irate chasers are desperately trying to halt your progress, and these need to be stopped.

John When playing, it soon becomes clear that the baddies move a lot faster than the Dinky Digger — which makes the securing of their destruction virtually impossible. 5/10

Tony The main fault with this game is that it's simply too hard. The poor presentation spoils the overall appearance of what is essentially a good idea. 6/10

Mark The idea behind this game is a good one but it gets lost somewhere, due to the unsophisticated and unpolished finished product. Nevertheless, it's playable and moderately enjoyable. 6/10



MAGIC MEANIES
CDS Microsystems/£5.95

This is one of those tunnelling maze games where the player's character — a wizard — makes his own maze as he goes. The idea is to defeat the magic meanies whilst picking up all the lead which can be transformed into gold.

Mark There are already a few games of this type, but this is one of the better ones. Nine speed levels are included, and these begin quite easily, increasing to a level where a high degree of manual dexterity is required. 8/10

John Colour and graphics, although not spectacular, are certainly adequate, giving a reasonable amount of enjoyment. The plot is rather silly, but the game's very addictive. 7/10

Tony The graphics used in this game are a little limited — because they're one graphic block in size, and move only one block at a time. 8/10



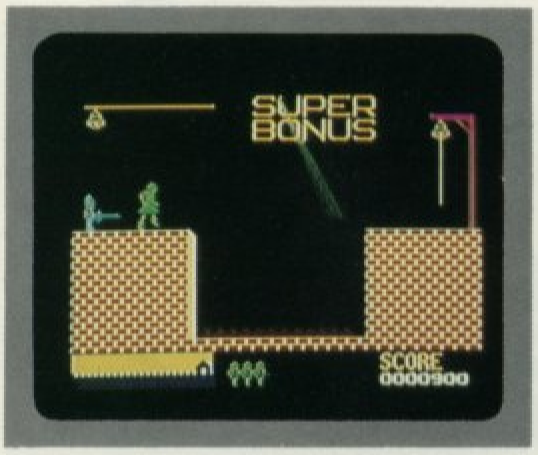
PINBALL WIZARD
CP Software/£5.95

The title really speaks for itself; a simulation of the old arcade favourite, and a rather complex one at that, complete with flippers, spring action firing, bumpers, and many others.

Tony This game is so well written that everything works as it does on a real pinball machine. The flippers move instantaneously, and the scoring makes excellent use of sound. 9/10

Mark The perfect choice of colour and superb graphics all go to make this a most amazing simulation. Ball movement is impressive and very realistic. 10/10

John A brilliant idea that's totally absorbing and so addictive it makes this game one of the best to have been put on the Spectrum. Only one thing: perhaps it would have been nice if a change of scenery had been included. 10/10



HUNCHBACK
Ocean/£6.90

Based on a classic theme, the player's job is to enact the part of Quasimodo and rescue the beautiful Esmerelda while avoiding the guards and fireballs, and having to jump over pits of fire. At the end of each screen Quasimodo must ring 'the bells, the bells'.

John 'Quasi' is quite easy to control, but the 3D effect sometimes makes it a bit difficult to judge when to jump and so on. Nevertheless, that's not really a problem, and the speed of the game is quite comfortable throughout. 8/10

Tony One really amusing thing about the game is that 'Quasi' looks like he's wearing high heels, and the guard seems to be doing a goose-step. 8/10

Mark One teeny weeny 'bug' seems to be present — 'Quasi' can't use the bridges the knight builds. Still, it doesn't spoil the player's enjoyment one bit. 9/10



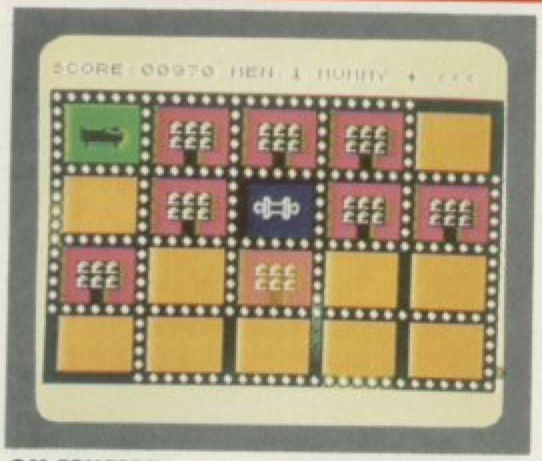
NIGHTFLIGHT II
Hewson Consultants/£5.95

This is another flight simulator package, but one that is able to fit into 16K. You take the controls of a 'Cessna'-type plane and negotiate a safe landing.

Tony Two outstanding features of this game are the speed at which it responds to your commands, and the choice of colour. Both are so good, that they make flying this plane less difficult than those found in other simulators. 9/10

Jon A very useful feature is the clear 3D representation of the runway; apart from this, though, nothing else can be seen from the cockpit as you're flying at night. 8/10

Mark The program offers as many as seven options; such as looking at your position or going into training mode. The colour is used well, but not extensively and the graphics are superb. It's also joystick compatible. 9/10



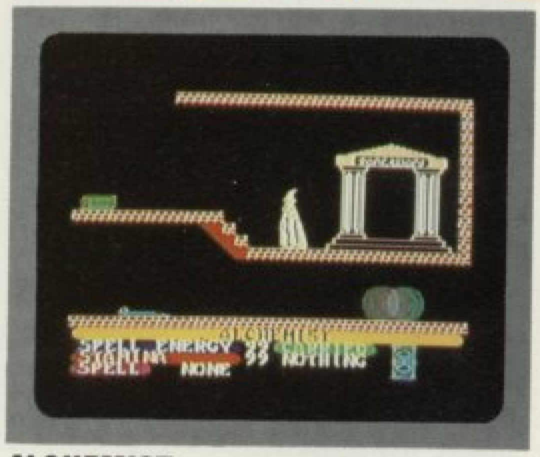
OH MUMMY
Gem Software/£5.95

You're on an expedition to the ancient Pharaoh's tomb, to control a party of explorers so that they bump into five royal mummies. You do this by finding keys, treasure and magic scrolls, all the time avoiding the tomb's guards.

John This game has five speeds which range from fast to virtually impossible. Although all the colours of the spectrum (no pun intended) are used, the overall effect is a sort of orange haze — making its use average but not spectacular. 6/10

Mark The guards and yourself are represented by simple block graphics which give little detail; however, the various artefacts are better defined. 7/10

Tony Despite the simple but pleasant little tune that's played at the start, the overall effect could have been far better. The picture on the cassette insert, however, is excellent. 6/10



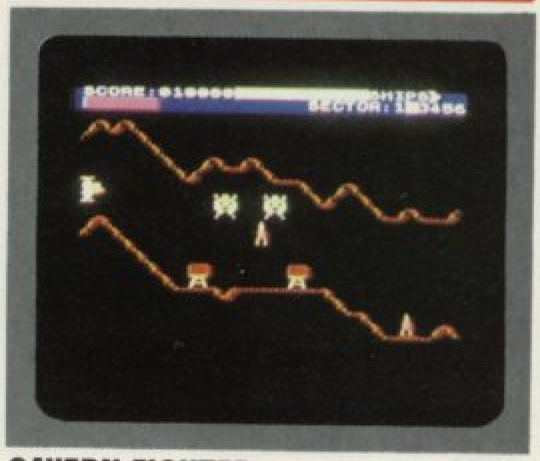
ALCHEMIST
Imagine Software/£5.50

A graphic adventure game where the player takes on the role of an alchemist, and must collect various objects and the four parts of the destruction spell to aid him in his quest to kill the evil warlock.

John This is certainly the most original game to come from Imagine. Its use of colour must have stretched the Spectrum to its limits (if not beyond), and includes the smoothest, flicker-free, high resolution graphics ever produced on this machine. 9/10

Tony The eagle looks like an eagle, the wizard's movement is realistic and the use of sound is excellent. There's only one small problem — the 'baddies' move too fast. 9/10

Mark The speed of this game is just right when you first play it, but seems a little slow as your expertise increases. Highly recommended. 9/10



CAVERN FIGHTER
Bug-Byte/£5.95

The aim of the game is to protect Watford from hostile attack *(Why bother? Ed.)*, and this is done by taking control of a Gamma 14 flight craft and winging your way through long narrow caverns, shotting out the enemy rockets' fuel dumps and destroying meteors.

Tony This is an incredibly fast game considering the amount going on, but the graphics aren't very appealing, neither is the choice of colour. 5/10

John It's a bit quick in the early stages, but this is far from a disadvantage. On the contrary, it quickly makes the player more skilful, and better prepared for the later stages. 7/10

Mark Colour is well used which makes the player's own craft, as well as the assorted aliens, stand out against the background. The graphics are smooth and fast. 8/10



JUMBLY
Dk'Tronics/£6.95

This is one of those slide puzzles, the kind that used to be found in Christmas stockings and crackers every year. You start with a picture grid, and all the squares jumbled up; your job is to put them back in order.

Tony The pictures generated are of a particularly high standard and use colour to the best advantage. But needless to say, they are generally static and, therefore, this is less of a plus. 6/10

John Having instructed the program to move one of the squares, it shifts to the new position very quickly. Sometimes the response to the player's commands is so good, the wrong section moves. 7/10

Mark The pictures generated are superb, and the speed of the program is very fast, giving a good response time — apart from when it's jumbling the letters which takes about 15 seconds. 8/10



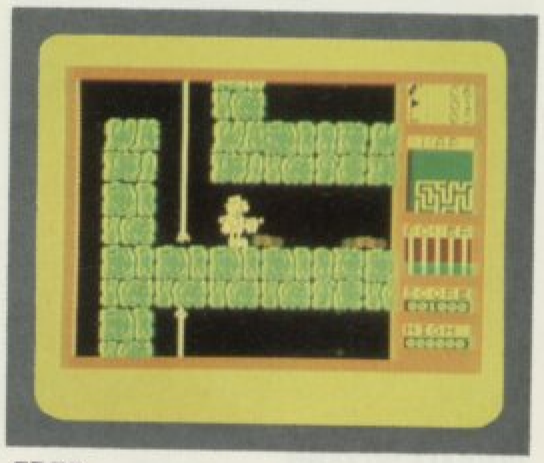
CLASSIC ADVENTURE
Melbourne House/£5.95

This is a copy of the first ever text adventure game which was written to run on a large mainframe computer. The aim of the game is to search for hidden treasure, which is often guarded, and then return it to your home base.

Tony The program responds almost immediately. And with your commands appearing in yellow and the program's messages in white, the game is fairly simple to play. 7/10

John This is a good conversion of the original which requires plenty of concentration and practice to be successful. The cassette insert is colourful and includes very comprehensive instructions. 8/10

Mark An adventure that has an extensive vocabulary, always keeps you interested and puts your brain to work. A good adaptation of the original. 9/10



FRED
Quicksilva/£6.95

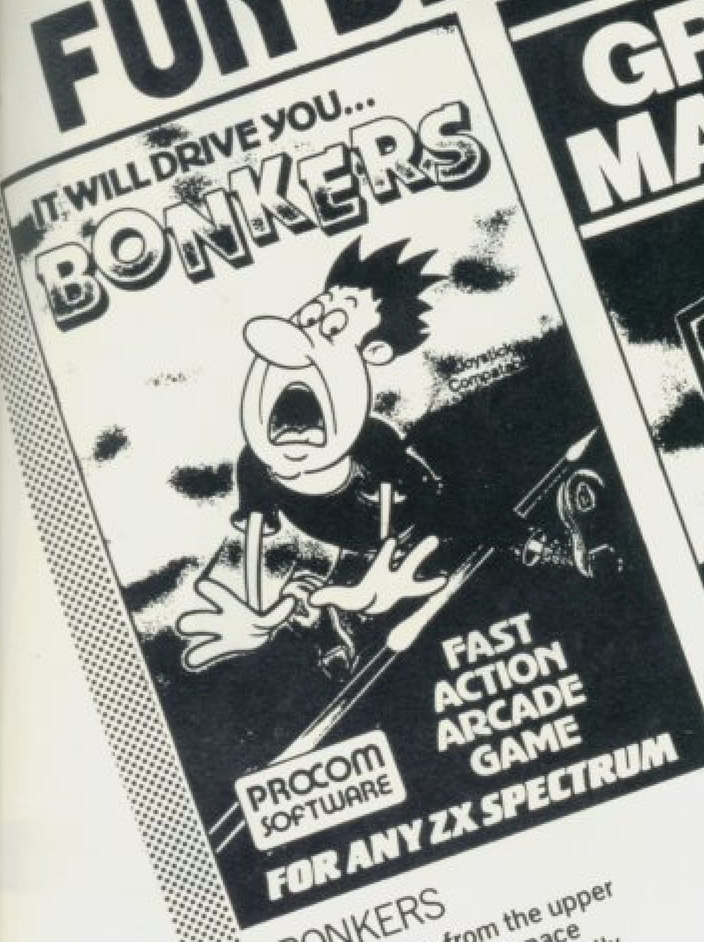
You are Fred, a fearless archaeologist, and must work your way round the pyramid in search of ancient treasure. It's not quite that simple as there are ghosts, vampires and acid drops which all try to hinder your progress.

Mark This is quite an original maze game that has an unusual twist. The speed is just about right — fast enough to keep you on your toes. 8/10

Tony A game which includes many novel twists. Its use of graphics and colour is very good and the overall appearance is just about perfect. 9/10

Jon This is not a particularly new idea for a game, but it does have superb graphics which are large and well-defined. The instructions on the insert, are very clear and precise, and this makes the game addictive, easy to play and very user-friendly. 8/10 YS

WHERE THE FUN BEGINS...



BONKERS

Try to escape from the upper regions of an alien space station . . . Fight the deadly Pac-Droids and Baby Droids — but do not relax — there are other aliens waiting!
* Arcade Action!
* High Score price!
for any ZX Spectrum

£5.50

DON'T LET THE SLOW START FOOL YOU!



GRID MASTER

A game for all the family to enjoy. A draw can never be achieved . . . or can it? Lots of fun!
for 48K ZX Spectrum

£4.95



BREAKAWAY

Be the modern Robin Hood! Time based, you have 15 seconds to rescue the damsel in distress . . . Can you do it? Try! Fun for all the family for ZX Spectrum 16K/48K

£5.50

FOR THE ZX SPECTRUM

PROCOM SOFTWARE

* Bonkers and Breakaway are available from Prism, Woolworths, Rumbelows, John Lewis and Debenhams.
* Dealer inquiries welcomed
* Overseas inquiries welcomed

DO YOU HAVE A GOOD IDEA FOR A GAME? CALL US!

COMING SOON!

* Bricky Micky
* Bermuda Triangle
* Time Bug
* Magic Orchard
* Final Frontier

ask for details . .

HOW TO ORDER

Telephone orders
(01) 508 1216

Credit Card Holders (Access/Barclaycard only) can purchase by telephone. Please give card no., name, address and items required, and your order will be despatched the same day.

BY POST

Simply write your requirements in the coupon provided, enclose your cheque/P.O. or use your Access or Barclaycard. Please make cheque payable to; PROCOM SOFTWARE and post to address below. Allow 7 days for delivery.

IN PERSON

Visit our premises, at the address below for a great selection — we will be delighted to see you.

PROCOM SOFTWARE

PROCOM SOFTWARE
309 HIGH ROAD · LOUGHTON · ESSEX
TEL: (01) 508 1216

POST TO: PROCOM SOFTWARE,
309 HIGH ROAD, LOUGHTON, ESSEX.

Please send me the following items:

☐ Bonkers (£5.50)
☐ Grid Master (£4.95)
☐ Breakway (£5.50 each)
☐ List of new games!

(Please tick items)

I enclose my cheque/P.O. for £ ______

I prefer to pay with my ACCESS/BARCLAYCARD
(Delete whichever not applicable)

Card No. ______

Signature ______

Name ______

Address ______

Tel: (Day) ______ Tel: (Eve) ______

Credit cards valid if signed by card holder.
Address above must be the same as card holder.

VS 2|3/84

THE YS TOP 20

Well, here it is . . . the *YS* Top Twenty! Obviously, there are any number of charts available to show the state of software sales and, if you look around, they all seem to be different. Funny, eh?

So what we've decided to do is to invite you the readers to tell us your current favourites; that way, in future issues we'll be able to publish a chart that really does go some way to reflecting your fave raves. At the bottom of the page you'll find a coupon for entering your top five tapes of the month. Either fill this in and send it to us, or pop off a postcard with your top five scrawled upon same.

As we collect your entries, every month they'll be punched into the nearest Speccy and out will come the *YS Reader's own chart*. That way, with any luck, we'll be able to see just how accurate all these competing charts are. And as just a weeny incentive to get you writing to us, each time we work out the chart, all the entries will be put in a *large* hat (belonging to our editor, of course) and the first out will receive three of the latest software releases for the Spectrum.

Don't miss out on our hopelessly inadequate prize. Send your selection today!

YS TOP 20 READER POLL

My top five fave raves on the Speccy are:

1 ____________________

2 ____________________

3 ____________________

4 ____________________

5 ____________________

Name ____________________

Address ____________________

I qualify for the prize, as a member of the human race.

Send this offer to:
**YS TOP 20
YOUR SPECTRUM
14 RATHBONE PLACE
LONDON W1P 1DE**

YOUR SPECTRUM

THIS MONTH		
1	**HUNCHBACK** OCEAN	A FAST-MOVING ARCADE CONVERSION THAT HAS YOU JUMPING THE WALLS OF THE CASTLE TO SAVE ESMERELDA.
2	**MANIC MINER** SOFTWARE PROJECTS	JOIN MINER WILLY IN HIS QUEST FOR THE MISSING KEYS, IN 20 BIZZARE AND COLOURFUL CAVERNS.
3	**ATIC ATAC** ULTIMATE	A FULL-GRAPHICAL ADVENTURE IN AN ARCADE FORMAT, BUT WATCH OUT FOR THE WANDERING GHOSTS!
4	**STONKERS** IMAGINE	A COMPLEX GRAPHICAL WAR SIMULATION WHERE YOUR TASK IS TO TAKE OVER THE WORLD — STRICTLY FOR MEGALOMANIACS ONLY.
5	**DEATHCHASE** MICROMEGA	REMEMBER THE 'BIKE CHASE' SCENE FROM RETURN OF THE JEDI? ALL HAS BEEN RECREATED
6	**CHEQUERED FLAG** PSION	JOIN THE GRAND PRIX CIRCUIT AND RACE (SMASH UP?) SOME OF THE GREATEST TRACKS IN THE WORLD.
7	**FLIGHT SIMULATION** PSION	A HARDY PERENNIAL. TAKE CONTROL OF THE AIRCRAFT IN THIS SUPERB 3D SIMULATION.
8	**ANT ATTACK** QUICKSILVA	AN AMAZING 3D ADVENTURE IN THE CITY OF ANTESCHER THAT HAS YOU RESCUING YOUR DUMB MATE FROM THE KILLER ANTS.
9	**MR WIMPY** OCEAN	A CHANCE TO DISCOVER THE HORRID TRUTH OF HOW BURGERS ARE MADE — AN ARCADE CONVERSION.
10	**ALCHEMIST** IMAGINE	A WIZARD PROGRAM WHICH MANAGES TO BLUR THE BORDERS BETWEEN ADVENTURE AND ARCADE.
11	**FIGHTER PILOT** DIGITAL SYSTEMS	A 3D FLIGHT SIMULATION PROGRAM THAT SITS YOU BEHIND THE CONTROLS OF A FIGHTER PLANE.
12	**LUNAR JETMAN** ULTIMATE	A COMPLEX ARCADE/ADVENTURE THAT TAKES PLACE ON THE FROZEN WASTES OF THE MOON.
13	**JET PAC** ULTIMATE	FIRST YOU BUILD A SPACESHIP, THEN YOU GO 'STAR-TREKKING' AROUND, COLLECTING GEMS FROM OTHER WORLDS.
14	**HORACE GOES SKIING** PSION	A GAME THAT BUNDLES THE CLASSIC FROGGER AND SKI SLOPE PROGRAMS INTO ONE.
15	**PYRAMID** FANTASY	THERE ARE 120 COLOURFUL ROOMS, AND ALL YOU HAVE TO DO IS COLLECT THE CRYSTALS — THE TROUBLE IS IT'S RAINING ALIENS!
16	**POOL** CDS	A SIMULATION OF THE GAME OF POOL — FOR SPORT AND PUB FANATICS ONLY.
17	**SPACE SHUTTLE** MICRODEAL	A 3D FLIGHT SIMULATION THAT HAS YOU CLAUSTROPHOBICALLY COSSETTED IN DEEPEST, DARKEST SPACE.
18	**BIRDS AND THE BEES** BUG-BYTE	AN ORIGINAL IDEA FOR AN ARCADE GAME THAT INVOLVES COLLECTING NECTAR WHILST KEEPING OUT OF THE WAY OF THE CENTIPEDES
19	**VALHALLA** LEGEND	A GRAPHICAL/TEXT GAME THAT HARKS BACK TO THE WORLD OF NORSE MYTHOLOGY FOR ITS INSPIRATION.
20	**HARRIER ATTACK** MARTECH/DURREL	ATTACK THE OCCUPANTS OF A SMALL ISLAND WITH THE VALIENT LADS OF THE HARRIER DIVISION.


3 from POSTERN for '84
XANAGRAMS
HANGMAN
EXCITE
ANAGRAM
PENGWYN
dinky digger
2
1
3
XANAGRAMS
Over 5,000 permutations in this game of skill for all ages.
(Spectrum 16K/48K BBC 'B' Electron CBM 64)
DINKY DIGGER
It's super-fast, furious and as mean as they come.
(Spectrum 48K)
PENGWYN
As the temperature rises, strange happenings begin in Pengwyn's frozen world.
(Spectrum 48K BBC 'B' Electron CBM64 Spectra video).
POSTERN
Available nationally from specialist retailers and all good multiples.
Please write quantities required in boxes provided and state computer type.
Name
Address
POST TO: Postern Ltd., P.O. Box 2, Andoversford, Cheltenham, Glos GL54 5SW. Postern is always on the look out for any new games you might have developed.
POSTERN
Total £ or Access No.
Please Postern People post me:
1 Xanagrams
2 Dinky Digger
3 Pengwyn
Price £6.95 each
(£7.95 for overseas mail order)

QL USER



A Nation Yawns...

QL delivery update

Editor Roger Munford; **Managing Editor** Bruce Sawford; **Deputy Editor** Tina Boylan; **Editorial Assistant** Peter Shaw; **Software Consultant** Gavin Monk; **Sub Editor** Nik Lumsden; **Contributors** Ron Smith, Max Philips, Andrew Pennell; **Art Editor** Jimmy Egerton; **Art Assistant** Steve Broadhurst; **Group Advertisement Manger** Jeff Raggett; **Advertisement Managers** Shane Campbell, Gill Harris, Jason Wood; **Production Editor** Derek Cohen; **Typesetters** Beverley Douglas, Maggie Kayley, Velma Miller; **Production Manager** Sonia Hunt; **Group Art Director** Perry Neville; **Publisher** Steve England; **Distribution Manager** Colin James; **Published by** Sportscene Specialist Press Ltd, 14 Rathbone Place, London W1P 1DE. Company registered in England. **Telephone (all departments)** 01-631 1433. **Telex** 8954139 BunchG. **Reproduction** Graphic Ideas, London; **Printers** Chase Webb Offset, St Austell, Cornwall; **Distribution** Seymour Press, 334 Brixton Road, London SW9. Telephone 01-733 4444.

 QL User is a monthly publication.

Q L N E W S

Supplying the Mythical with the Hypothetical

In the race to produce hardware add-ons for the as yet unseen Sinclair QL and, of course, gain the distinct commercial advantage of being first, manufacturers are still falling over each other trying to get their hands on either a machine, or a detailed technical specification. And interestingly enough, lack of information hasn't stopped at least two companies from advertising their hypothetical wares.

The first of these is Miracle Systems of Cambridge, which is planning to supply the QL owner with a parallel printer interface. "But", we asked Mr Honeyball, "how can you produce such a device without the necessary technical details?".

"Well, actually" he said "it's not in production yet. But it'll only take us three or four days to tailor the interface after the first batch of QLs become available". Anyway, for £49 the purchaser gets an interface that plugs into the serial port, and one that doesn't require any software to drive it. With any luck it should be available some time in April. More details on 0223 312886.

Xcom (Services) Ltd, of London, has started advertising a QL compatible floppy disk drive, and this will be supplied with a controller that plugs straight into the expansion slot. But until Xcom can get a QL to play with, customers will have to make do with an RS232 interface, says the company's Mr John Seaman

There is, however, one company claiming to have a 'friend' at Sinclair Research who has been able to supply not only information, but QLs for limited periods of time. This has apparently enabled the company to copy the ROM, disassemble it, work out what's going on and design add-ons accordingly. But sad to say, all this effort has so far come almost to naught. Each different machine popped through the back-door has been modified to overcome certain inherent problems. All of which, says our source, suggests that Sinclair Research is still changing the hardware. Our Cambridge mole is also able to confirm rumours that the Basic is so full of 'bugs' that it keeps crashing, and that the manual still hasn't been completed. Were all this to be true, we might be seeing delivery dates nearer to Christmas than Spring! **RS**

The Great QL Accumulator

The latest official news from the Sinclair Research camp would seem to indicate that QLs will not have been despatched until the end of March has gone. The company had originally scheduled delivery of the first machines to customers for the end of February, but due to delays "caused by the final stages of development taking longer than planned", the opening date has been put back a month.

Attempts to discover the reason for the switch from "phenomenal demand" to "development problems" to explain away the delay were met with frosty replies that amounted to little more than "no comment". Also parried were questions on current rumours that suggest the company has no QLs in stock (at time of writing) and that the SuperBasic ROM was still in the design stage when the device was launched.

Sinclair Research has already sent out letters to all who've staked their claim for one of the wonder machines, explaining the delay and informing them (roughly) when to expect delivery. For those who ordered their machine at the launch, the letter proclaims hopefully that "we expect to be able to deliver your QL not later than the end of May". However, the grapevine would seem to suggest that orders were placed in a *very* large hat and picked out at random — something which has not pleased many of the software houses who biked over their orders to Sinclair Research HQ, even before the dust had settled from the launch.

> If you have applied for QLUB membership, we will be sending your membership card with your QL, and your twelve month period of membership will commence at that time. During your membership you will receive 6 issues of the QL newsletter, the first of which will be sent shortly after your QL.
>
> The demand for the QL has been phenomenal from the day we launched it. We expect to be able to deliver your QL not later than the end of May. We realise that the time between now and then will be frustrating, but we are confident that your QL will be worth waiting for and, of course, we will do everything possible to beat our target date for sending it.
>
> Yours sincerely
>
> Nigel Searle
> Managing Director

Part of the original 'dear John' letter sent out by Sinclair Research.

Although Sinclair Research is being fairly tight-lipped about the state of play, the word is that the company's Camberley offices are receiving about 500 orders a day — which works out at something like a staggering £200,000 daily!

Assurances have been made by Sinclair Research that none of your precious money will be frittered away while you're sitting there awaiting the postman's knock. The cash is reported to be stashed away in a trust fund so no-one can get their hands on it — but just who will reap the benefits of the interest is anyone's guess. The only consolation for those who've shelled out handfuls of the green folding stuff is that Sinclair Research is thinking of giving the late receivers a 'free' gift to compensate for the wait — although according to a Sinclair Research spokesperson, "we don't know what it will be — that has yet to be arranged". Sounds interesting!

So, if you'd like to enter your name in the QL 'tombola', you'll probably get a free something or other. However, if you want to hang on to your money until your QL is despatched, you'd be better advised to order via a credit card — that way no money will change hands until delivery is due.

Around the Houses



HUNT EMERSON

With tongues wagging throughout the industry,* Ron Smith, *software sleuth, tracks down the rumours to their source and tries to answer the question on everybody's lips — just who has got a QL?

Fired with enthusiasm by the spectacular launch of the new Sinclair QL, software houses wasted no time in placing their orders for one (or 10!) of Sir Clive's wonder-machines. Naturally they were keen to start writing all manner of new and interesting programs to keep the new QL owner suitably amused. That dream soon got lost amidst the mountain of unfilled orders and discarded production schedules. However, still undaunted, *YS* decided to track down any brave (foolhardy?) pioneers out there, eager to be first in the great QL race — when it begins that is!

OUT IN THE COLD

Paranoia seemed the order of the day. Most of the software producers we talked to all seemed to think that some independents had some kind of special relationship with Sinclair Research that enabled them to receive early machines, they being left out in the cold. This was echoed by John Fletcher of PSS who said "There are perhaps one or two software houses who might have a QL, but we certainly don't". He did add, however, that when a machine did arrive they would consider putting something together.

A little more optimistic, if just a touch confused, was Carol Hewlett from CDS Microsystems. When asked if CDS had any plans for producing QL software she said, "We're definitely writing software for the QL, but at this stage I can't tell you what it is". Was it going to be business or games software? Even education, perhaps? No good, Ms Hewlett wouldn't be drawn. So what about the machine itself. Do you actually have a QL, we asked. "Yes, I think so," she said. A strange answer, considering the vast number of people who would give an arm and a leg — if not other more valued parts of their anatomy — to have one available. We asked Ms Hewlett to check her stock of QLs. She said she would and promised to phone back. We're still waiting.

GLIMMER OF HOPE

Just in case CDS are, in fact, QL-less (and good news anyway for all those who want to know what's happened to their orders), Carnell's Stuart Galloway offers a glimmer of hope. He phoned Sinclair Research at the beginning of March to enquire what had happened to his machine; his cheque having already been cashed. He was told by a "pleasant young lady" that the first batch of QLs were going out the very next day. She didn't tell him to hold his breath — and he hasn't. These delays, however, don't worry Mr Galloway. Two weeks before the machine was launched, he signed up a programmer "who knows the 68008 like the back of his hand", and has got to the point where he's completed a utility program on paper —all he needs now is a machine to test it on!

Equally interested, though less concerned with trying to write software at present, is Bug-Byte. Their Tony Baden feels that "it could well be Christmas before the QL is in the shops", and if that does turn out to be the case, as he says "there's very little point in rushing". Another stumbling block, according to Baden, is the lack of a cassette interface. This, he feels, could be a serious disadvantage, especially when coupled with the apparent lack of QL Microdrives.

SPREADING GOSSIP

The doom-laden and negative comments continued with Program Power's Chris Payne who said "we will be writing software for the QL, when we can get hold of one. But for the time being, we'll just have to wait". However, plans do seem to be afoot, because Chris says "we'll be approaching QL software development from both angles, ie. business and games". He also didn't miss the opportunity to perpetuate a few rumours, by saying "I'd heard the Basic hadn't been finished yet" which should please the technical staff at Sinclair Research.

Spreading gossip was obviously the order of the day because Mr Payne proceeded to let slip that his colleague was "one of those Cambridge (or was it Oxford?) dons, who's done plenty of work on the 68000 chip" (although the gentleman in question, John Haig, might not like that to be too widely known).

Both Mikro-Gen and Quicksilva felt that their existing close links with Sinclair Research had had absolutely no effect on them getting an early machine. In fact, Mikro-Gen's Mr Denial (*Who he? Ed.*) kept very quiet, and would only say that "it's early days yet — we've nothing planned". And Quicksilva's Paul Cooper told us that they were "having lots and lots of talks with Sinclair Research, but it doesn't appear to be having any effect". It's the same old story of a software house with "plans and ideas for QL software" who will just have to wait "until one turns up". Mr Cooper was also in the mood for joining the gutter grapevine. He said "I heard rumours at the LET show that machines wouldn't be available until the third quarter" — which may well turn out to be nearer the truth than many would like to think.

Artic, like everyone else, is just waiting to get its hands on a machine. Their Margaret Turner remarked that the "plan is to transfer some of our Spectrum software to the QL".

A GOOD LEAD

There is, of course, one company in the land that's had no trouble at all laying its hands on a machine — and that's Psion. Talking to Charles Davis about the immediate future, he said "We're continuing to produce software for the QL, in addition to the four packages already announced. But it would detract from the impact of future releases if we were to talk about them now". He did mention, however, that *Flight Simulation* would be available in the near future. But what about independent software producers? Mr Davis again: "My view is that everyone will be producing software for the QL, but we have a good lead, as we've got some machines". Lucky for some!

Soft Sells

The old story that software sells computers is taken to heart on the QL where the four Psion packages that come with the machine provide serious business tools at a home micro price. Quentin Lowe has a look at the software that maketh the machine . . .

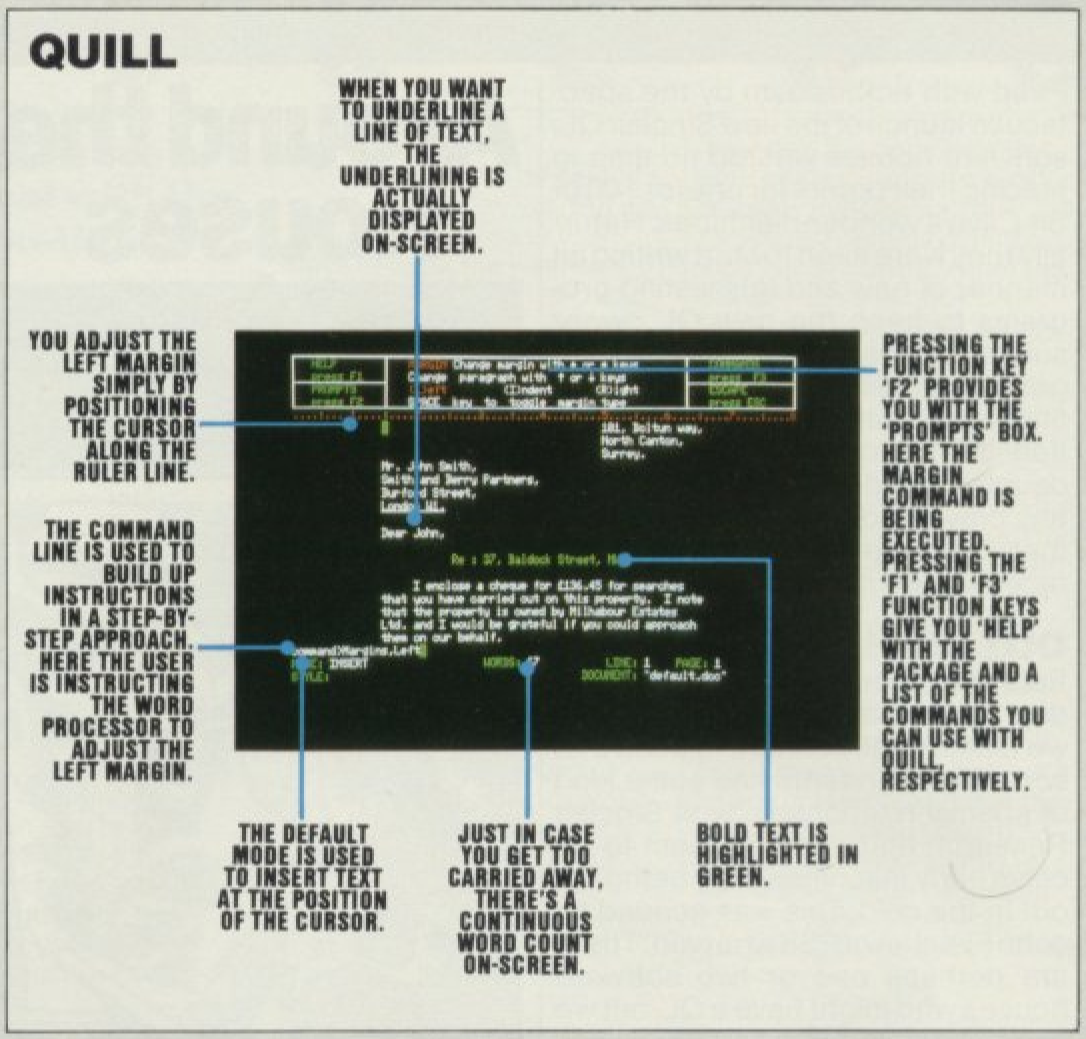
QL QUILL

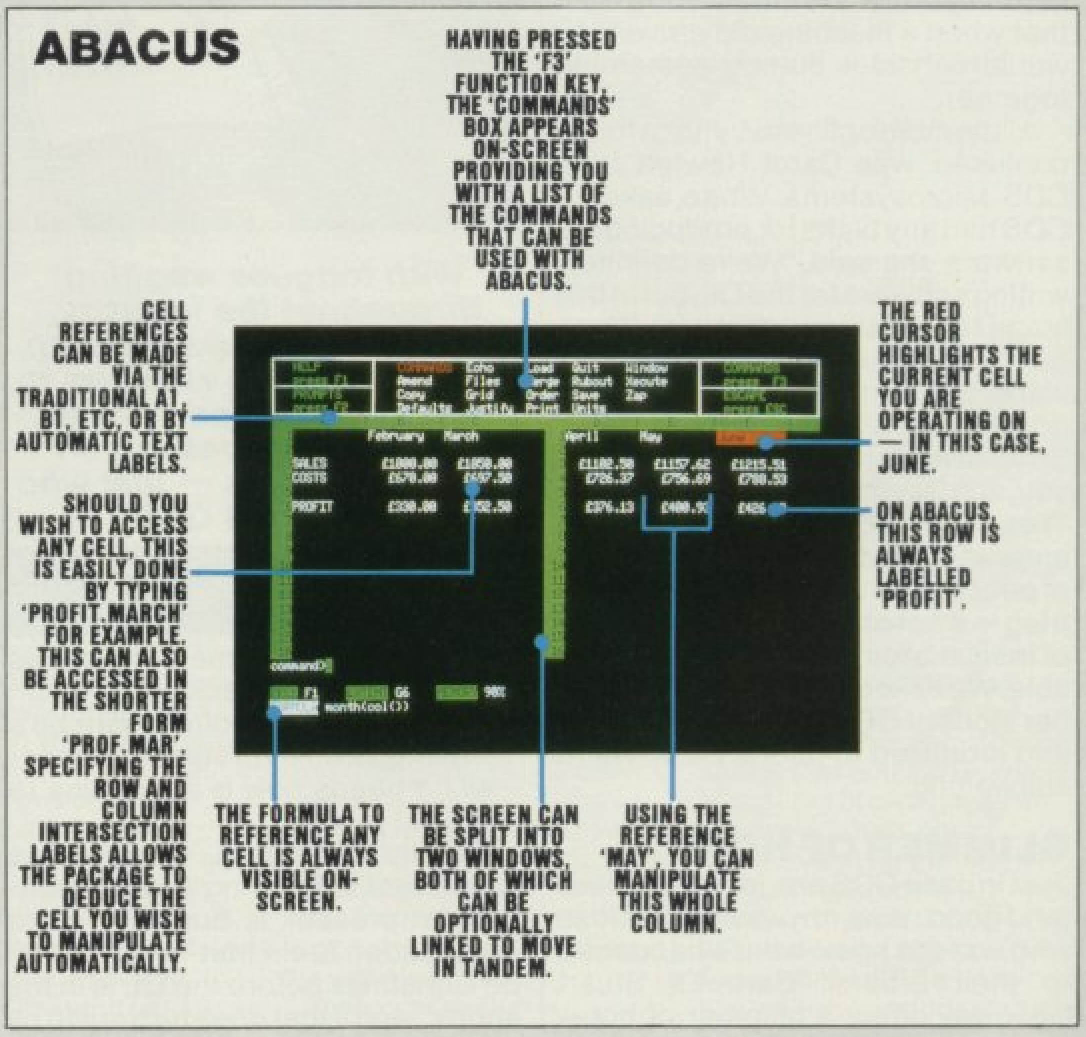
The word processor *QL Quill* is going to be the predominant influence on QL sales. For a mere £400 plus the cost of a printer, it allows the QL to be used as a serious small word processing tool — breaking, for many newcomers, a genuine price barrier.

Quill is a serious grown-up word processor — text appears on screen in the same format it will be printed — and if you want justified text, it's justified as you type. Page breaks occur visibly on the screen, underlined text is underlined and bold text is highlighted in a different colour. Forgetting all the arguments about what is and what isn't wanted in an editor, this system makes *Quill* very simple to learn and use.

There are all the usual features — search and replace, block copies and moves and so on. You also get the luxury of a glossary feature — any frequently needed phrase (or set of commands!) can be assigned to a particular letter key ready for instant recall. *Quill* even saves the glossary to its Microdrive between sessions so it's always there when you work.

It's most serious limitation is one which affects many small machine packages — the maximum amount of text that can be edited at any one time. With *Quill*, it's likely to be around ten pages — enough for letters, reports and so on but perhaps a restriction for some people. Even so, there's no doubt that *Quill* is good enough for its challenging role.





QL ABACUS

Abacus provides a serious spreadsheet package on a par with top selling programs like *Multiplan* and *VisiCalc*. Although *QL User* has yet to have the chance to perform formal benchmarks, the program is very responsive in use. What's more, Psion seems to have actually introduced one or two sensible and welcome extensions to the facilities normally available in spreadsheets.

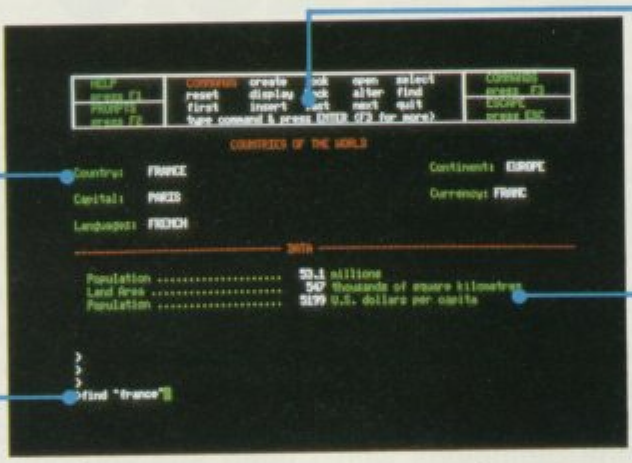
The first is cell labelling. Traditionally, each cell in a spreadsheet has been named by co-ordinates, such as A1, B2 or R3C4 and so on. Some newer packages have had a labelling facility where you can give cells sensible names. With *Abacus*, all text in the sheet can be used as labels — the program automatically deduces which cells to use when you use a label in a command.

And *Abacus* is extremely good at text as well as numbers. There are lots of functions dealing with character information (such as INSTR, CODE, UPPER, LOWER and so on), making it easy to generate really comprehensible and easy-to-use

ARCHIVE

ALONG WITH THE ARCHIVE PACKAGE, YOU GET A SAMPLE FILE CALLED GAZETTEER — THE ONE THAT HAS ALWAYS BEEN PART OF PSION'S VU-FILE PROGRAM FOR THE SPECTRUM.

OF ALL THE PACKAGES SUPPLIED WITH THE QL, ARCHIVE IS THE ONLY ONE THAT IS COMMAND-LINE DRIVEN. YOU CAN EITHER USE THE COMMANDS ALREADY BUILT INTO THE PACKAGE OR YOU CAN DEFINE YOUR OWN PROCEDURES.



PRESSING THE 'F3' FUNCTION KEY ALLOWS YOU TO SEE ALL THE COMMANDS YOU MAY WISH TO USE ON ARCHIVE. THESE ALLOW YOU TO USE THE QL AS A SIMPLE CARD INDEX.

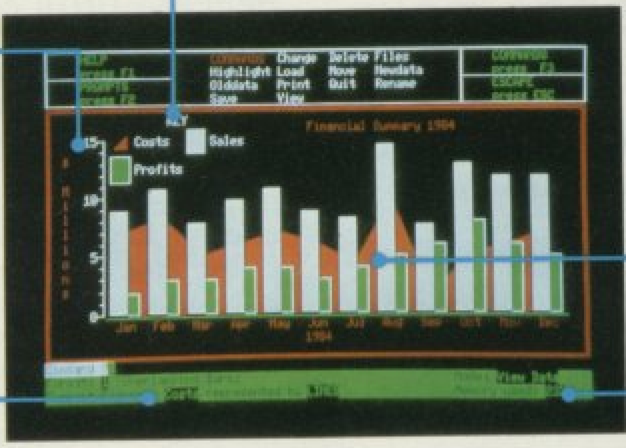
HERE THE SCREEN LAYOUT USED IS THE ONE THAT IS SET UP BY DEFAULT BY THE PACKAGE ITSELF. HOWEVER, YOU CAN ALSO ACCESS THE DATABASE WITH A NUMBER OF DIFFERENT LAYOUTS OF YOUR OWN DESIGN.

EASEL

YOU WILL AUTOMATICALLY GET A KEY TO THE GRAPH ON-SCREEN AS A GUIDE TO WHAT EACH COLOUR REPRESENTS.

ACCORDING TO THE DATA YOU PROVIDE THE PACKAGE, THE AXES ARE AUTOMATICALLY SCALED AND

THIS IS AN INDICATION THAT THE CURRENT SET OF DATA BEING WORKED ON IS THE 'COSTS'. SALES AND PROFITS HAVE ALREADY BEEN DEALT WITH AS SEPARATE ITEMS OF DATA AND HAVE BEEN PLOTTED ON-SCREEN. SHOULD YOU WISH TO MANIPULATE DATA ALREADY ON-SCREEN, YOU CAN EDIT THEM USING THE 'OLDDATA' COMMAND.



THE GRAPH SHOWN IN THE SCREEN SHOT IS AN OVERLAPPED BAR CHART ILLUSTRATING SALES AND PROFITS, COMPLETE WITH THE COSTS SUPERIMPOSED AS A LINE GRAPH FILLED TO THE X-AXIS.

ALL THE PACKAGES PROVIDED ON THE QL UTILISE THE 'PERCENTAGE MEMORY LEFT' DEVICE, POPULARISED BY MICROSOFT. THIS GIVES YOU A NON-TECHNICAL INDICATION OF FREE MEMORY.

models. This ability should open up many more applications for *Abacus*, making it one spreadsheet that's suitable for more than just accounts.

QL ARCHIVE

Archive is a heavyweight database package with the added attraction that it can be operated on a simple level with no more trouble than a trivial program (such as Psion's own *Vu-file*) on the Speccy. To do this, Psion has gone for a now traditional card index format, with each entry in the database effectively being its own 'card' and being divided up into separate fields of information.

A set of simple commands have been added to let you introduce new records, search for information and so on. And the clever bit is that you then discover a whole new programming language hidden away inside *Archive*. It's at a higher level than Basic — with powerful English-like commands and a special intelligent editor that takes care of program layout for you. *Archive* has commands that let you have a number of different screen layouts with any one file, combine different files and so on.

The result is that you can customise *Archive* exactly to a particular application, in much the same way as packages such as *dBase II*. All that power does make it the hardest of the four packages to learn, although you can get into it very easily just by starting off with a simple card index. However, compared to most products of a similar capability, even the most complex of *Archive* procedures is relatively easy . . .

QL EASEL

Easel is one of those packages that's good for a quick thrill — a sort of 'what can it do then?' program that gives instant and dramatic results. It's designed to produce business graphics — bar charts, line graphs, pie charts and so on — and in an area that's still relatively new to computing, *Easel* is likely to be something of a pioneering program.

Most business graphics packages are turgid programs that are either too complex to be practical or too restriction to be of use. Easel is really easy. It's based around complete named sets of data — which may have been typed in, imported from another program (*Abacus* perhaps!) or even derived from existing sets of data using *Easel*'s built-in simple calculation facilities. Once in, you can graph any of the data in any way you like — mixing different sets on the same graph, tipping the graph sideways, customising the design of the bars and background right down to carefully adding special labels, and positioning the key where you want it. You can also make further cosmetic changes to the graph, such as colour alterations, solid outlines and 'filled-in' areas.

All the time *Easel* provides sensible suggestions for things you've not specifically detailed. It scales and labels the axes and provides a key; you can also do as little or as much work as you like on achieving the perfect final product. Compared to other top business graphics packages, *Easel* doesn't offer any dramatic new abilities. What it does provide, however, is a system that's so quick and easy to use that you don't mind messing about getting the best result — an ideal tool for producing effective graphs in the minimum of time.

Holy Structures... It's SuperBasic

Zap, Pow, Blam . . . just when you thought it was safe to get back to unstructured waters, along came the Cambridge Joker in his QL-Mobile with a surprise present — SuperBasic — to threaten an imprecise programming world.* Andrew Pennell *battles to explain . . .

The subtly named SuperBasic on the QL appears to have interesting features, particularly its ability to use Pascal-like structures, similar to BBC Basic. I say 'appears' because at the time of writing (some weeks after the 28-day expiry) no QLs have been manufactured; this article is based mainly on information given in the provisional *User Guide* and therefore no guarantees can be given as to its total accuracy.

Having thus 'copped out', let's first of all consider the subject of structured programming. Different people have different views, but for me it's programming in such a way as to be 'self-documenting' — that is, you're able to work out what a section of program does without too much effort. The much maligned GO TO and GO SUB statements have long come under attack by purists — certainly they make program flow more difficult to follow on paper. The QL manual says that the extra SuperBasic commands make these statements superfluous, and there's a lot of truth in it. When I was taught Pascal, I remember the lecturer putting a total ban on the GO TO statement, which horrified me at the time. But she was right — you don't need it in Pascal, and you shouldn't need it in SuperBasic, if you use the control structures efficiently; none of the QL programs in this article have it.

There are four major enhancements to bog-standard Basic in the QL that relate to structured programming — functions, procedures, REPEAT loops and the SELECT command. They are very similar to certain Pascal statements, except that SELECT in SuperBasic is known as CASE in Pascal. I shall endeavour to show you each of them in use, and compare them to their 'equivalent' in Spectrum Basic.

LOOKING AT FUNCTIONS

QL-type functions are a great improvement on Spectrum types, allowing as they do multi-line definitions and the use of local variables. On the Spectrum, DEF FN statements could only be followed by an expression, with no other statement types allowed. Any attempts at recursion (ie. the function calling itself) resulted in an 'Out of memory' error, after a delay while the machine stack filled. SuperBasic functions, on the other hand, do allow recursion, as well as single-line definitions with the normal syntax.

An example of a recursive function is shown on Listing 1a, where a function called 'fact' calculates the factorial of a number. (The factorial of a number is the result of all integers up to and including it being multiplied together — eg. factorial 4=1x2x3x4=24.) Lines 1010-1020 check first for factorial zero, which by definition is '1'. It does this by using the QL IF...THEN...ELSE commands, and RETURN 1 is the way to return a value of one of the function. If it's not zero, then lines 1030-1040 make the function return a value calculated from another factorial, hence the recursion.

> *"The QL manual says that the extra SuperBasic commands make the GO TO and GO SUB statements superfluous"*

The Spectrum version is shown in Listing 1b and it has several disadvantages when compared to the QL version — and that's apart from its lack of structure. In particular, the variable i is corrupted by the routine, and the 'input' variable must always be a, and the 'output' variable is always fact. With the QL version, there are no variable restrictions — if you wish to make zz equal to the factorial of b, then you could use zz=fact(b).

As you can see, to use a QL function you don't need the RN of other Basics — you just use it like any of the built-in functions.

Listing 2a shows another QL function, which is used for a 'live' string input routine called 'getstring$'. It shows several of the QL's features, the first of which is LOCAL. If you want to use variables in a function definition (or in procedures, shown later) and you don't want them to affect any variables in the rest of the program, then the use of LOCAL will ensure that the values of any existing variables with the same name are preserved through the function or procedure, then restored afterwards.

Another 'structured' feature is the REPEAT command, used for looping. Basically, if you want a loop of some sort, put 'REPEAT somename' at the start, and 'END REPEAT somename' at the end. Then, where you want to put a test to leave it, use 'IF condition THEN EXIT somename' and the named loop will no longer execute. Unfortunately, this is not very similar to either Pascal or BBC Basic's REPEAT...UNTIL construct, making conversion more difficult.

The final, rather neat, feature used is the SELECT statement, which allows easy choices to be made without unwieldly IF...THEN statements — something which has not been implemented on any previous Basic. Line 1010 ensures first that any variables called a$ or b$ in the calling program are not affected, and then it enters the main loop, 'getloop'. The two REPEATs in lines 1030 and 1040 are *short* forms of the statement, and execute the multi-statements that come after them automatically without END REPEAT commands; they're used in the function to scan the keyboard.

Lines 1050-1170 consist of SELECT statements for taking a number of different actions, depending on the key used. If Newline (CHR$ 13) is pressed, then the main loop, 'getloop', is left. If backspace (CHR$ 8) is pressed, then (if allowed) a backspace is printed and a character removed from the end of b$. Note that QL string handling is conducted in the same (non-standard but neat) way as the Spectrum and ZX81. If it's a non-control character (lines 1120-1160) it's printed, then added to b$.

Finally line 1190 ensures that the value returned from the function is b$.

For reference, the Spectrum equivalent is shown in Listing 2b and it displays similar disadvantages to Listing 1b. I've made one assumption in the QL version, and that is that backspace is CHR$ 8 (it should be as it's the ASCII standard code).

As well as functions, the other major structure addition is that of procedures. A procedure is a sequence of instructions, optionally using parameters, which is basically just an upmarket GO SUB. They each must have a different name, and are invoked by simply using their names, unlike fussy BBC Basic which requires PROCname. This allows extra commands to be added, without resort to the machine code that's required when adding commands to the Spectrum — and the BBC Micro for that matter.

In Listing 3a, a QL procedure called 'box' is defined which, not surprisingly, draws a box that's defined by its bottom left-hand corner, its width and height. The QL has an improved version of DRAW, which can cope with many parameters, each separated with TO. To use it, for example, to draw a box at (10,20), size 300x200, is simply: box 10,20,300, 200.

Note that unlike DEF PROC no brackets are required when the procedures are actually used. The Spec-

SuperBasic-A Spectral Comparison

```
1000 DEF FN fact(a)
1010 IF a=0 THEN
1020  RETURN 1
1030 ELSE
1040  RETURN a*fact(a-1)
1050 END IF
1060 END DEF
```

Listing 1a

This is an example of recursion, where the function 'fact' calculates the factorial of a number. Note the use of QL keywords IF... THEN... ELSE.

```
1000 LET fact=1
1010 IF a=0 THEN RETURN
1020 FOR i=1 TO a
1030 LET fact=fact*i
1040 NEXT i
1050 RETURN
```

Listing 1b

The Spectrum Basic version of Listing 1a — you can see that there are a number of variable restrictions, ie. the variable i is corrupted by the routine.

```
1000 DEF FN getstring$
1010 LOCAL a$,b$
1020 LET b$=""
1025 REPEAT getloop
1030  REPEAT getpause: IF INKEY$="" THEN EXIT getpause
1040  REPEAT getkey: LET a$=INKEY$: IF a$<>"" THEN EXIT getkey
1050  SELECT ON CODE a$
1060   ON CODE a$=13
1070    PRINT : EXIT getloop
1080   ON CODE a$=8
1090    IF b$<>"" THEN
1095     b$=b$( TO LEN b$-1)
1100     PRINT CHR$ 8;" ";CHR$ 8;
1110    ENDIF
1120   ON CODE a$=32 TO 127
1130    IF LEN b$<32 THEN
1140     b$=b$+a$
1150     PRINT a$;
1160    ENDIF
1170   END SELECT
1180  END REPEAT getloop
1190 RETURN b$
1200 END DEF
```

Listing 2a

This program example illustrates a 'live' string input routine called 'getstring$'. Notable QL features to watch out for are the LOCAL, REPEAT and SELECT commands.

```
1000 LET b$=""
1010 PAUSE 0: LET a$=INKEY$
1020 IF CODE a$>127 THEN GO TO 1010
1030 IF a$=CHR$ 13 THEN PRINT : RETURN
1040 IF a$<>CHR$ 12 THEN GO TO 1100
1049 REM backspace
1050 IF b$="" THEN GO TO 1010
1060 LET b$=b$( TO LEN b$-1)
1070 PRINT CHR$ 8;" ";CHR$ 8;
1080 GO TO 1010
1100 IF LEN b$=32 OR CODE a$<32 THEN GO TO 1010
1110 LET b$=b$+a$
1120 PRINT a$;
1130 GO TO 1010
```

Listing 2b

The equivalent Spectrum Basic to Listing 2a, although seemingly shorter, is somewhat lacking in structure. The problem of variable restriction is also present.

```
1000 DEF PROCbox(x,y,width,height)
1010 PLOT x,y
1020 DRAW x+width,y TO x+width,y+height
1030 DRAW x,y+height TO x,y
1040 END DEF
```

Listing 3a

This QL procedure draws a box by defining its bottom left-hand corner, width and height. More complex parameters may be included, separated by the keyword TO.

```
1000 PLOT x,y: DRAW width,0
1010 DRAW 0,height: DRAW -width,0
1020 DRAW 0,-height
1030 RETURN
```

Listing 3b

Spectrum Basic allows the routine in Listing 3a to be written in a simpler fashion — it uses a DRAW statement that is relative rather than the QL's absolute equivalent.

```
1000 DEF PROCpformat(b)
1010 LOCAL a,a$
1020 a=(INT (b*100))/100
1030 IF a<.01 THEN PRINT "      0.00";: RETURN
1040 IF a>1e6 THEN PRINT "      ?.??";: RETURN
1050 a$=STR$ a
1060 IF a$(1)="." THEN a$="0"+a$
1070 IF a$(LEN a$-1)="." THEN a$=a$+"0"
1080 IF a$(LEN a$-2)<>"." THEN a$=a$+".00"
1090 REPEAT strlen
1100 IF LEN a$=9 THEN EXIT strlen
1110 a$=" "+a$
1120 END REP strlen
1130 PRINT a$;
1140 END DEF
```

Listing 4a

This procedure is suitable for a financial package in which you wish to right-justify a series of numbers on the decimal point. Again, use is made of the LOCAL and REPEAT commands.

```
1000 LET a=(INT (a*100))/100
1010 IF a<.01 THEN PRINT "      0.00";: RETURN
1020 IF a>1e6 THEN PRINT "      ?.??";: RETURN
1030 LET a$=STR$ a
1040 IF a$(1)="." THEN LET a$="0"+a$
1050 IF a$(LEN a$-1)="." THEN LET a$=a$+"0"
1060 IF a$(LEN a$-2)<>"." THEN LET a$=a$+".00"
1070 IF LEN a$<9 THEN LET a$=" "+a$: GO TO 1070
1080 LPRINT a$;
1090 RETURN
```

Listing 4b

To simulate Listing 4a in Spectrum Basic, you come up against the same problem experienced in Listing 1b — namely, that of variable restrictions, ie. you lose the old value of the variables, a and a$.

trum version, in Listing 3b, is a bit simpler than the QL version because it uses a relative DRAW statement, whereas the QL's is absolute.

Listing 4a shows another QL procedure called 'pformat', which prints a number right-justified on the decimal point and therefore it's for printing currency. It's probably called 'pformat' because format is a reserved word; there's already a Basic statement with that name. It uses a LOCAL statement, as well as a REPEAT loop to finally right-justify the output string, a$. It uses the STR$ command, found on most other Basics but missing from the SuperBasic manual. Because of the variable coercion on the QL, it may well be possible to replace it with: 1050 a$=a.

The equivalent Spectrum program is shown in Listing 4b, and its main disadvantage is again the loss of the old value of variables, a and a$.

The use of procedures greatly enhances the readability of programs if

"SuperBasic has some interesting features, particularly its ability to use Pascal-like structures, similar to BBC Basic"

used properly and, with sensible names. As an example:

```
100 print_instructions
110 setup_game
120 play(speed)
130 do_highscore
```

is much more understandable than a possible Spectrum equivalent of:

```
100 GO SUB 9000
110 GO SUB 2300
120 GO SUB 4200
130 GO SUB 1200
```

The way that functions and procedures add features to QL Basic is extremely useful, so much so that on switching on your QL, it will ask you if you want to LOAD any from Microdrive.

I hope this has given an insight into the 'structured' aspects of SuperBasic on the QL; you can all practice now while waiting for one to be delivered. Who knows, if it really catches on, it could be the end of the GO TO for good!

SINCLAIRWATCH

BY GUTTERSNIPE

Well, Sinclair Research has done it again — not only has it launched the world-beating QL, there also looks like a near record-breaking delivery delay to go with it.

At the time of the QL launch, assurances were made that the first customers should get their machines by the end of February, and that the 28-day promises on the adverts would be stuck to. Mind you, the launch wasn't without its sceptics, especially when it was discovered that the Microdrives weren't fully working under QDOS — and that not one single machine was available for review. Indeed, it transpires that the Microdrive problem was the reason why the four Psion programs shown were stored elsewhere, and not Sir Clive's tricky little tapes.

Late Again!

The climaxing anticipation had burnt out to a whimper by the end of February. Not a single customer had come even within a sniff of receiving a QL and indeed Sinclair Research hadn't even made enough to allow review machines to be sent out. Unhappy punters received a letter giving reasons for the delay — 'phenomenal demand' it said, but just how demand for a new product can be so great as to halt production is a little puzzling.

The more experienced Sinclairologist among us may remember hearing this particular excuse before — it was also used to explain away the delays for the ZX Spectrum, the ZX Printer, the 16K RAM Pack, the ZX81, the ZX80 and even the Mk 14 — in other words, every single Sinclair Research computer product. Using the same excuse for the same inexcusable delay shows at the very least a profound lack of imagination.

The tragedy is that the QL disappointment (and, in retrospect, the Spectrum one too) was easily avoidable. If Sinclair Research had launched, but not taken any advertisements (or for that matter, printed any order forms) until the production line was genuinely cranking into action, then customers might have been a lot happier with the odd delay or two. Certainly, it would have been a tonic for Sinclair Research's market image. And as such as the company is quick to assure that the millions of pounds worth of cheques are going to be placed in a trust fund until the transactions are about to take place — in asset terms it's almost as good as money in the bank.

Just how long the delays are likely to be is difficult to assess, but those who ordered their QLs on the launch day have been told they should now get them by the end of April — a nail-biting wait of over three months. The less fortunate ones (*Guttersnipe* included) who ordered a few days after the launch, and those who replied to the lavish colour ads have been told that the wait could be as long as the end of June. At the current rate, with orders flooding in (at a reported rate of 500 a day) and QLs trickling out, the BBC Micro-style six month delays look all too imminent.

Chip Chat

What of the QL itself? Although with the absence of real machines no-one can be absolutely sure, it looks amazingly impressive — in fact, almost identical to the machine predicted in *YS*'s *Sinclairwatch* in issue one, with a real keyboard, dual processors and two Microdrives. The lesser eight-bit processor is the Intel 8049, commonly found inside dot-matrix printers, while the main chip is the Motorola 68008, a cut-down version of the immensely powerful 68000.

At the QL launch, Sir Clive was heard to state that "we waited for the 32-bit chip to get the extra memory". Needless to say, the 68008 only addresses 1Mbyte of memory (unlike its kith and kin in the 68000 series which addresses 16Mbytes) — exactly the same as any good ol' 16-bit chip, such as the 8088. Still, the argument over whether it's a '32-bit' chip will rage for some time yet, but Sinclair Research is the first outfit to term any member of the 68000 '32-bit'. Even its manufacturers refer to the 68008 as a '16-bitter'. The keyboard is thankfully pretty good, although with slightly non-standard layout — particularly the symbols above the numeric keys.

The Microdrives on the QL are improved Spectrum types, with the specification printed in the early Spectrum adverts. Some argue that Microdrives are not the ideal storage media for a business machine, not just because of their slow speed and lack of true random access, but because of the cartridge cost and the problems involved in their mass-duplication.

You Want It When?

The news of the QL launch probably came as a piece of bad news to those working for the Advertising Standards Authority; it's by no means unknown for the ASA to step in over the company's advertising of new products. But it seems the authority's protests in the past have had *some* effect back at the ranch; this time, the company contacted them first, explaining the problems over the 28-day clause on their adverts.

The result is that the latest ads say "Delivery may take longer than 28 days". This self-evident statement may perhaps ease the situation somewhat, though the ultimate cure would be for Sinclair Research to stop advertising for a while.



A rare shot of someone getting their hands on a QL (courtesy of Fantasy Photographs Ltd.).

ONLY THE BEST BECOME A....

FIGHTER PILOT

48K SPECTRUM

BY D.K.MARSHALL



A SPECTACULAR FLIGHT SIMULATION OF THE WORLD'S MOST EXCITING JET FIGHTER WITH STUNNING 3D COCKPIT VIEW AND THE MOST REALISTIC 3D AIR-TO-AIR COMBAT EVER SEEN ON THE SPECTRUM!

The SUPREME SIMULATION... it leaves the rest at the runway threshold!

OTHER AMAZING FEATURES

* Landing and take off from 4 runways plus steering on the ground.
* Fully Aerobatic Performance
* Airfield attack by the enemy
* Cross winds, Turbulence and landing in fog.
* 4 Pilot ratings, 8 Beacons, ground features and map
* Artificial horizon, radar and flight computer.
* ILS, Flaps, U/C and system failures
* 100% machine code with complete instructions and pilot notes.

£7.95

inc. VAT & p. & p.
Joystick Compatible
(Kempston, AGF, Sinclair Interface 2, Protek etc.)*



"Definitely the best simulation for the Spectrum..."
- Crash Micro





TAXI
£5.50
Any Spectrum
Become a Taxi Driver and test your skills - but beware the Reckless Drivers!
Fun for all the family



FOOTBALL
£7.95
48K Spectrum
League Analysis - Easy to use Pools Prediction.
With 9 methods of team performance analysis - let your Spectrum guide you to your pools selection.



NIGHT GUNNER
£6.95
48K Spectrum
Joystick Compatible *
The air battle where only aces survive - with enemy fighters attacking from all directions, guns blazing, explosive 3D ground attack sorties - and there's **still** the flight home.

WIZARDS OF COMPUTER GAMES SOFTWARE

DIGITAL INTEGRATION

Watchmoor Trade Centre,
Watchmoor Road,
Camberley, Surrey GU15 3AJ

Selected products available from:
Boots, Woolworths, H.M.V., Menzies, Greens, Lasky's, Currys, Rumbelows, Selfridges, Spectrum Dealers, Video Palace, Zappo, and all good Computer shops.

Please send me the following games for the Spectrum

Fighter Pilot 48K	☐ £7.95
Night Gunner 48K	☐ £6.95
Football 48K	☐ £7.95
Taxi (Any Spectrum)	☐ £5.50

Trade and Export enquiries welcome.

VAT and p. & p. inclusive within UK
(Overseas inc. 55p. per cassette)

Lifetime Guarantee Cheques payable to Digital Integration
I enclose a cheque/P.O. for________ Total.
Name________
Address________

Y.S.3

Or debit my Access/Account No.

Access

Tele Sales Camberley (0276) 684959

Please send to Digital Integration, Dept
Watchmoor Trade Centre, Watchmoor Road, Camberley, Surrey GU15 3AJ



BEAR
BOVVER
It's no
picnic!
artic
Computing Ltd

Home computers, they tell us, are *the* ideal vehicle for education — but what exactly supports this claim? *Sandy Dewhurst*, computer studies schoolteacher, sifts through the pages of various literary texts devoted to the subject.

A TOUCH OF CLASS

Let's get something straight right from the start. I have never read *Mindstorms*, that wise book of computer education by Seymour Papert. What's more, I have no regrets. I did take it with me on a 32-hour flight to Sydney a couple of years ago. After wading through the section about his childhood obsession with cogs and gears, quite frankly I decided to opt instead for the in-flight movies. As a classroom teacher, I'm more interested in texts of a less 'celestial' nature.

The first, and possibly most important thing an educational computing text must be is 'reader friendly' (sorry — just my computer humour). If it's a book for children, there needs to be the degree of informality needed to communicate with children on *their* level rather than the more usual artificial adult-imposed level. This requires a careful blend of both style and vocabulary. If the book is for teachers it needs to be clear and succinct. Busy teachers have little time for the wordy ramblings of computer enthusiasts.

One unfortunate aspect of many texts is that it's not altogether clear who they are for. The situation in computing today is such that the distinction between teacher and student is not as clearly defined as in other subject areas — and this is particularly so in primary schools. In many cases the pre-knowledge of the student is greater than that of the teacher. I don't intend to apologise for the authors of computing texts, but this must lead to confusion on their part. Far too often they counter this situation by simply writing a bland general text, with no real idea (or indication) of *who* should be reading it. Some books dealing specifically with the Spectrum are included in this category. A typical example of this is the Spectrum publication in the *Learning to Use...* series. In the publisher's own words, it is for "... potential users, established users, teachers, students and businessmen ...". Is there anyone else? I think that just about covers the entire population of the world (except business*women*!). In fairness to author Robin Bradbeer, I don't think it was ever written specifically for the education market; but it does demonstrate how general many of the books are.

A clearly defined purpose is essential for any text and for some this is more easily done than for others. Two books I looked at simply contained listings of programs — hence providing that clear and very specific purpose. *Educational Programs for the Spectrum* by Ian Murray and *40 Educational Games for the Spectrum* by Vince Apps both provide a wide variety of programs. They also include quite extensive notes on each program, giving detailed program descriptions, explanations of how to play each game, programming hints and even some educational notes!

However, their use is limited by their very specific purpose. I've often questioned the validity of books containing nothing other than listings — surely it would be quicker and easier for everyone concerned if the programs were put on cassette and sold as a software compendium. Someone in Century Publishing must have had similar thoughts, as cassette versions of the programs in Ian Murray's book are now available. The problem is, of course, the chances are you don't know this until *after* you've bought the book!

The listing in Ian Murray's book have obviously been printed, via an interface (see *Getting Into Print*, *YS* issue 2), on a professional printer. Unfortunately, a ZX Printer has been used to provide the listings in the other book, and several others I've looked at have done the same. The reproduction from such printouts is not up to a satisfactory standard for book publishing. It's a great pity that publishers haven't seen fit to obtain quality printouts of program listings. A little extra money in production would make quite a difference to the end product. In many cases the listings are illegible, and therefore totally useless. This is particularly the case when inverse video characters are used in the listing.

ON TO PROGRAMMING

Another area of specific focus for texts is programming. Most books in this field



NICK DAVIES

are supposedly designed with the student in mind. The first thing a text of this kind has to do is *not* speak down to the reader. Shiva has recently published *Programming for REAL Beginners — Stage 1 & 2*, which although not specifically saying 'educational text book', certainly implies it through the presentation. The major failing of these two is the fact that they are not machine specific and therefore there's a limit to the level the reader can progress with her/his own machine. Again, this is a case of trying to keep everyone happy, including the publisher's bank manger.

The presentation has been quite carefully planned — I was going to commend it until I saw the cartoon on page 78 of Stage 2. Pardon my high horse but it's sexist and not at all relevant to the rest of the text. The less obvious one on page 82 of Stage 1 is equally bad. Publishers should not be able to get away with such rubbish, and their use in the classroom should be limited.

Books focusing specifically on programming for children originated in the United States. Although good in their simple step-by-step development of programming skills, they lost a lot through their 'americana' style presentation — what I generally call 'gee whizz books'. Fortunately, they tended to be specific to American machines (Tandy, Apple, etc), so Britain was shielded from them to a large extent.

One final point on programming books — do they have to be in book form at all? One company, EDU-CAL, markets programming *workcards*, but unfortunately only for the BBC machine. A workcard is much easier to handle at a keyboard than a book, and it might well be worth investigating the possibilityfor the Spectrum (publishers, take note!).

THE NON-SPECIFICS

So, now let's look at the general educational books, which are neither program nor programming specific. What should they contain? How should they be presented? The short answer to these questions is, I don't honestly know. They all seem to follow a similar pattern, in both content and presentation. But, whether this is because the perfect formula was discovered early on which made an obvious model for everyone to follow, or because it's easier to follow previous published patterns with minor alterations to present a degree of 'individuality', who's to know. I have my own opinion on this matter, but I'll leave it up to you, the consumers, to decide for yourselves.

Whichever of these opinions you believe, it's unfortunate that the result is a lack of diversity in computing books in education. It's even more unfortunate when one considers the wide variety of texts available in other areas of educational publishing. One of the few publishing variants is Usborne, a company that, following on from similarly presented books in other fields, has produced bright, reader-friendly books on several subjects within the computing field. These are packed full of colourful illustrations — something that's all too frequently absent from other books in this field.

One book which didn't impress me at all to begin with was *Learning with your Computer* by Susan Curran and Ray Curnow (one of the *Clear and Simple* home computer series). However, subsequent readings have convinced me otherwise. There's a brief section on the background to educational computing and anything of a highly technical nature has been omitted, although not to the detriment of the text. If anything, this proves a positive move which gives the reader an informative but general overview of a complex subject. It's all that anyone, apart from a real enthusiast, would want.

One very important section in this book is the chapter on software selection where again the friendly, yet informative approach has been adopted. One thing though, all the suggestions offered involve obtaining information on programs from other sources — for example magazines and user groups. There's no effort made to advise readers on how to assess software themselves. The 'high priests' syndrome lives on: 'We are the chosen few, we have the knowledge, and we're keeping it to ourselves'!

The one factor that would keep it off a Spectrum user's shopping list is that ol' recurring problem, namely, that it isn't Spectrum specific. In fact, although it is a general computing text, all the program listings are for the Dragon 32. All right, so they only need minor adaptations to get them working on other machines, but again, what busy teacher has time for this sort of activity?

THE SPECIFICS

The Spectrum is quite a unique machine, so teachers (and students), really require books specific to it, and publishers must realise this. For this reason, I will finish by looking at two recently published books which attempt to fulfil this criterion, both being written by people highly respected in the computing world. The two books are *Spectrum in Education* by Eric Deeson, and *Educational Uses of the ZX Spectrum* by Tim Hartnell, Christine Johnson and David Valentine. The former is from Shiva's *Friendly Micro* series and the latter is the companion volume to Tim Hartnell's other Sinclair computerguide, *The ZX Spectrum Explored*.

Both titles mirror each other quite closely, even to the point of having some identical chapter titles. The first few chapters of each provide some basic ideas on computers and their educational applications, and these are followed by an introduction to programming. Eric Deeson tends to develop this to a higher degree, although there are two very useful chapters on the use of graphics in Tim & Co's book. In fact, they probably achieve the same results, programming-wise, by developing programs in specific subject areas of the curriculum.

Space is also devoted to the actual use of micros in schools. Eric's book focuses on the theory and historical context of micros in schools, whereas Christine Johnson contributes an enlightening couple of chapters in her book on the use of micros in a much-neglected part of the education system — the infant school. Another useful area in *Educational Uses . . .* concerns itself with software evaluations. It's useful because as well as all the previously mentioned sources of information, it actually explains how, without much effort, teachers can evaluate software themselves. So, there's no need to consult a 'high priest' after all!

I've always found a good guide to the quality of a computer book is to decide whether I would have it *off* my shelf. I say 'off', because books like these need to be used, not left *on* shelves. Top of my usability list would be *Educational Uses of the ZX Spectrum*, simply because it is the book *I* would find most useful as a primary teacher. If I was a high school teacher, however, I might well find the approach in *Spectrum in Education* more suited to my needs. I think I would also have *Educational Programs for the Spectrum* on my list, or better still, write off for cassette versions of all the programs.

Apart from these, there is very little that I could honestly recommend. It looks like education is the loser once again. I really look forward to the time when educational computing authors break free from their chains and follow the lead of some of their counterparts in the software field.

By the way, do you know what *CAF* is? It's worth buying Eric Deeson's book just to find out. His views on the future are very interesting. Read them. You may never view computers (or classrooms!) in the same light again

WE LOOKED AT...

Educational Uses of the ZX Spectrum by Tim Hartnell, Christine Johnson & David Valentine, Sinclair Browne, ISBN 0-946195-14-5, £6.95.
Spectrum in Education by Eric Deeson, Shiva, ISBN 0-906812-29-1, £6.50.
Educational Programs for the Spectrum by Ian Murray, Century, ISBN 0-7126-0260-7, £5.95.
40 Educational Games for the Spectrum by Vince Apps, Granada, ISBN 0-246-12233-1, £5.95.
Learning to Use the ZX Spectrum by Robin Bradbeer, Gower, ISBN 0-566-03481-6, £5.95.
Learning with your Computer by Susan Curran & Ray Curnow, Windward, ISBN 0-7112-0335-0, £4.95.
Programming for REAL Beginners — Stage 1 by Philip Crookall, Shiva, ISBN 0-906812-37-2, £3.95.
Programming for REAL Beginners — Stage 2 by Philip Crookall, Shiva, ISBN 0-906812-59-3, £3.95.

CALPAC LEARNING SERIES

Colour graphics and sound are used extensively to make learning more enjoyable.

5. MULTIPLICATION TABLES. This program with correction sequences enables you to test individual multiplication tables, or all tables up to a maximum that you choose. Carriages are added to a train as the score increases – see the train glide through the countryside when the test is finished.

6. THE ROMANS. Answer questions on the Romans to win a horse, chariot, viaduct and temple. Answer enough correctly and the horse will gallop away with its chariot.

7. SPELLING TESTER. The words in the test are initially displayed on the screen. Then short sentences are used as prompts for the words, which must by typed correctly to add stages to a space-ship.

Replace the words and sentences in the program with weekly spelling lists, save them on cassette, and you will rapidly create a whole library of programs.

8. HOMOPHONES. Sale or sail, bare or bear? Choose the correct word to complete the sentence. Extra motivation is provided by a coloured pot-plant, which grows with each correct response. The plant flowers in an unusually tuneful fashion at the end of the program.

J. J. Warren 1983

Published by Calpac Computer Software, 108 Hermitage Woods Crescent, St Johns, Woking, Surrey GU21 1UF

The programs on this cassette are subject to copyright, and may not be copied, transmitted, hired or lent without the written permission of Calpac Computer Software.

CALPAC

CALPAC LEARNING SERIES ZX SPECTRUM VOL. 2

CALPAC

CALPAC LEARNING SERIES

PROGRAMS 5-8



5 MULTIPLICATION TABLES
6 THE ROMANS
7 SPELLING TESTER
8 HOMOPHONES

SPECTRUM 16K 48K
FROM 7 YEARS

CALPAC LEARNING SERIES

Colour graphics and sound are used extensively to make learning more enjoyable.

9. NOUNS, VERBS, ADJECTIVES AND ADVERBS. Sentences are displayed and you select the named parts of speech. Instant feedback and helpful comments ensure positive learning. Correct answers are rewarded by the creation of a coastal scene, which is animated at the end of the program.

10. VERB PRACTICE. You have to complete the sentences using the correct tenses of the verbs. The program concentrates on those irregular verbs that often cause difficulty. An underwater landscape is created as questions are correctly answered.

11. THE STRUCTURE OF THE FLOWER. This program explains how the parts of the flower are involved in the formation of seeds. This is a three part program which makes full use of high resolution colour graphics.

12. LONG DIVISION. This detailed program takes the learner through long division sums in easy stages. Correction sequences are automatically provided when they are needed. Sums with remainders can be chosen if required.

J. J. Warren 1983

Published by Calpac Computer Software, 108 Hermitage Woods Crescent, St Johns, Woking, Surrey GU21 1UF

The programs on this cassette are subject to copyright, and may not be copied, transmitted, hired or lent without the written permission of Calpac Computer Software.

CALPAC

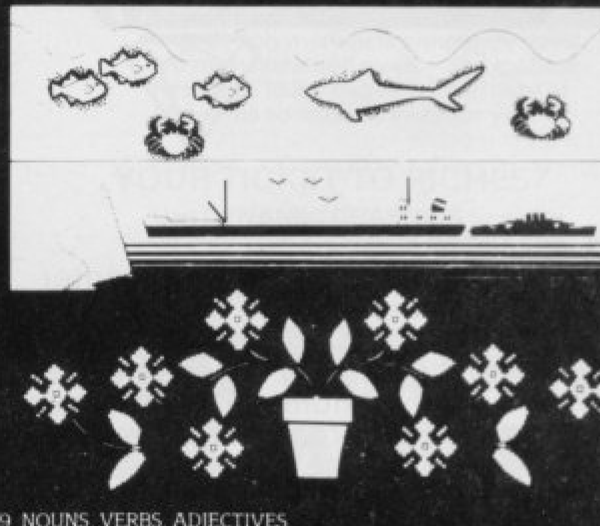
Our software is available by direct mail from:-
CALPAC COMPUTER SOFTWARE
108 Hermitage Woods Crescent
St Johns, WOKING, Surrey GU21 1UF

CALPAC LEARNING SERIES ZX SPECTRUM VOL. 3 FROM 9 YEARS

CALPAC

CALPAC LEARNING SERIES

PROGRAMS 9-12



9 NOUNS, VERBS, ADJECTIVES AND ADVERBS
10 VERB PRACTICE
11 THE STRUCTURE OF THE FLOWER
12 LONG DIVISION

SPECTRUM 16K 48K
FROM 9 YEARS

Additional features of the CALPAC LEARNING SERIES include:-
*Spelling checkers
*"Help" call up routine
*Easy insertion of subject material of your own choice into the programs

Our software is suitable for use in the home, school or college.

For further details or retail stockists please telephone:- 048 67 2584

We have a demonstration cassette available for retailers or schools.

We would like to hear from good programmers, graduates or teachers wishing to participate in the expansion of our Learning Series on the Spectrum, BBC Model B and other machines.

Our software is availabe by direct mail from:–
CALPAC COMPUTER SOFTWARE
108 Hermitage Woods Crescent
St Johns, WOKING, Surrey GU21 1UF

PRICE LIST

CALIPAC LEARNING SERIES VOL 1	16K or 48K Spectrum	☐ £9.50
CALIPAC LEARNING SERIES VOL 2	16K or 48K Spectrum	☐ £9.50
CALIPAC LEARNING SERIES VOL 3	16K or 48K Spectrum	☐ £9.50
CALIPAC CHEMISTRY SERIES VOL 1	48K Spectrum	☐ £7.50
CALIPAC CHEMISTRY SERIES VOL 1	16KZX81	☐ £7.50
CALIPAC PATTERNS	16K ZX81	☐ £5.95

FOR FURTHER INFORMATION PLEASE COMPLETE COUPON BELOW

NAME ....

ADDRESS ....

....

AVAILABLE FROM HMV STORES & SELFRIDGES

ADDING ZIP!

▶ 58 (roughly) the approach used by the language Forth. It also produces short code and is faster than p-code (since addresses rather than code are used), but there are still lots of instructions which aren't really needed — an extra GO SUB or CALL for every code, plus operations to fetch and store data between routines (you can't just leave it in registers since temporary results would get overwritten). A library of subroutines must be present when the program is run.

The last technique is to produce a lump of 'in-line' machine code for each intermediate code. The result is a long, fast program. The lumps of code (which we'll call templates) are read from a table and then patched if need be — modified slightly so that they reference the correct data. It would be daft to have different templates for every value or variable name, so we use generalised templates and POKE the copy so that it refers to the right data.

ZIP uses a mixture of these last two techniques. Where possible, templates are used for each operation or where the operation is slow or complicated calls (such as INPUT or CLS) to library routines are used — though only where the overhead of calling makes very little difference to the execution time. The *ZIP Instructions* panel lists the ZIP intermediate code instructions in code order (so VARiable FETCH is code 1, RETURN is code 6, and so on). Instructions are marked by an '*' if a library call is used.

Notice that words such as INVERSE, OVER and so on occur in two places. The first is for their use as statements on their own, where they change the effect of all subsequent screen use — the second is for their use within PRINT, PLOT or DRAW statements, where they only affect items drawn in the current statement. The special code PR-OPEN is used to mark the start of a new PRINT statement (so that the temporary colours can be discarded). The last two codes — PLOT and DRAW — will be added to the compiler in Part 3 of this series, as an example of the way ZIP can be extended.

THE LIBRARY

One problem with using a library is that there are inevitably errors in it, and whenever you correct part of it the addresses of subsequent routines change. If the Basic of ZIP contained the addresses of templates and library routines this would be very annoying, since every 'bug-fix' would require changes to a whole set of constants. ZIP doesn't work that way (if it did, we'd probably never have finished it!).

The library starts with a list of addresses, one after another. The first is the address of the 'VARiable FETCH' code, and so on. The list is produced by the assembler, as the 800-odd lines of library assembler are converted to machine code. To fix a bug in any template or routine you simply re-assemble the corrected code, and all of the addresses are put right. You can modify the library without altering the Basic, since it just uses simple offsets into the list — '4' for GO TO, '5' for GO SUB, etc, and copies code from the address indicated in the list at position '4', '5' or whatever. The *Library 'Vector' Table* illustrates this.

The program at the end of the listings can be used to enter the library data. The table contains checksums (modulus 256 totals of every eight bytes) so there's a check for typing errors after each eight values. The left-hand column is the address (in decimal) and should not be entered.

AND NEXT . . .

Besides the Basic listing of ZIP, next month we'll discuss the process of compilation in detail, with help from programming guru *Jon Smith*. If compilers confuse you, machine code is a mystery and your Spectrum is sluggish, don't miss ZIP next issue! YS

THE FIRST IN DYNAMIC PACKAGES FROM PROCOM



WHAT IT IS

Whether you are a beginner or an experienced programmer it will make no difference with the DYNAMIC GRAPHICS package from Procom. You will be able to program professional looking graphics on your Spectrum. DYNAMIC GRAPHICS comes in two cassettes at the price you would normally pay for one. But do not let the low price put you off. DYNAMIC GRAPHICS is by far the most unique and much needed software aid on the market today.

£14·95
inc. VAT and P&P.

JUST LOOK AT WHAT WE HAVE IN STORE FOR YOU!

TAPE 1

THIS IS A CHARACTER DESIGNER WITH A DIFFERENCE * FULL SCREEN ANIMATION OF SPRITES * 27 FUNCTIONS INCLUDING SPRITE HANDLING 96 CHARACTERS WHICH CAN BE SAVED ON TAPE TO USE IN YOUR OWN PROGRAMS * EASY METHODS OF INCLUDING CHARACTER DESIGNS IN YOUR OWN BASIC PROGRAMS * NO KNOWLEDGE OF MACHINE CODE NEEDED * SCREEN SAVING * SPRITES MAY BE TREATED AS SUPER LARGE CHARACTERS (4 x 4). THE LIST IS ENDLESS. YOU WILL BE ABLE TO DESIGN LITERALLY THOUSANDS OF DIFFERENT CHARACTERS FROM PAC MEN TO SPACE MEN OR SPACE SHIPS TO BATTLESHIPS.

TAPE 2

THIS IS THE ULTIMATE UTILITY FOR COMPOSING HI-RES GRAPHICS ON YOUR SPECTRUM. AS WELL AS ESSENTIAL HAND DRAW CURSOR CONTROLS (OR JOYSTICK) THERE ARE MANY ADVANCED FEATURES, SUCH AS FAST FILL AND RESCALE. THERE ARE IN ALL OVER 50 COMMANDS. A CONSTANTLY UPDATED STATUS DISPLAY. GENERATED PROMPTS ARE FULLY DISPLAYED. NOW YOU CAN DESIGN SCREENS AS GOOD AS THE SOFTWARE COMPANIES.

1. DYNAMIC PROGRAMMING
2. DYNAMIC ANIMATION
3. DYNAMIC SOUNDS
4. DYNAMIC FILING
5. DYNAMIC EDITOR & ASSEMBLER

Package for the ZX Spectrum now! (Also for the BBC, Electron and CBM 64)

EASY TO USE

DYNAMIC GRAPHICS is the one that everybody will understand. The instruction manual is simplicity itself. No more strained eyes — the manual has large 10 x 8 pages with plenty of diagrams to make it easy for you.

YOUR TICKET TO RICHES?

Use DYNAMIC GRAPHICS to make up your own program, and if it's good we will market it for you and maybe make you rich.

DYNAMIC GRAPHICS

PROCOM SOFTWARE

**309 HIGH ROAD
LOUGHTON
ESSEX
Tel: (01) 508 1216**

HOW TO ORDER

**Telephone orders
(01) 508 1216**

Credit Card Holders (Access/Barclaycard only) can purchase by telephone. Please give card no., name, address and items required, and your order will be despatched the same day

BY POST

Simply write your requirements in the coupon provided, enclose your cheque/P.O. or use your Access or Barclaycard. Please make cheque payable to; PROCOM SOFTWARE and post to address below. Allow 7 days for delivery.

IN PERSON

Visit our premises, at the address below for a great selection — we will be delighted to see you.

Post to: PROCOM SOFTWARE, 309 HIGH ROAD, LOUGHTON, ESSEX.

Please send me the following items:

☐ DYNAMIC GRAPHICS ☐ DYNAMIC PROGRAMMING ☐ DYNAMIC ANIMATION
☐ DYNAMIC SOUNDS ☐ DYNAMIC FILING ☐ DYNAMIC EDITOR & ASSEMBLER

(Please tick items)

I enclose my cheque/P.O. for £ ____________

I prefer to pay with my ACCESS/BARCLAYCARD
(Delete whichever not applicable)

Card No. ____________
Signature ____________
Name ____________
Address ____________

Tel: (Day) ____________ Tel: (Eve) ____________

Credit cards valid if signed by card holder.
Address above must be the same as card holder.

YS284

SUBCONSCIOUSNESS



Fantasise over features . . .



Marvel at machine code . . .



Revel in reviews . . .



Lust over listings . . .

If you recognise any of the critical conditions portrayed above, then you're probably a lost cause as far as computing goes — admit it, you're hooked! And, as a devout follower of Sir Clive's wonder machines, you don't want to miss out on one of the best cures around (our magazine carries no Government health warning).

You've doubtless heard the rumour that you'll always be able to take a stroll down to your local newsagents and pick up a copy of *YS*, no problem? But the fact is that *Your Spectrum* issue one sold out so fast, we're still getting calls from people anxiously pulling their hair out trying to secure a copy. Really!

So, that's why this month, we're kicking off our *Speccy Subs* service. For just £12 (in UK and Northern Ireland only — it's a little bit more for Europe and Air Mail) you can guarantee that each issue will come thudding through your door each and every month.

All you have to do is fill in the attached coupon (or a photocopy of same) and unlock your chequebook. Of course, if you're a credit card king, just give us your number and we'll do the rest — that'll do nicely.

YS PRIORITY ORDER FORM

I would like to subscribe to *Your Spectrum and QL User* for the next 12 issues, starting with the .... issue.

I enclose a cheque/postal order made payable to Sportscene Specialist Press Ltd for (tick the appropriate box):

☐ £12 (UK and Northern Ireland)
☐ £15 (Europe) ☐ £25 (Airmail)

Or please charge my Access/Visa/Diners/American Express card (please delete where applicable).

Account Number ______________________

Name: ______________________

Address: ______________________

____

____

______________________ Postcode ______________

(please use block capitals)

Send the completed coupon plus payment to **Speccy Subs, Your Spectrum, 14 Rathbone Place, London W1P 1DE.**

If you don't want to deface your magazine please make a photocopy of this order form.

AGF

PROGRAMMABLE JOYSTICK INTERFACE for Spectrum or ZX81

ONLY 26·95 +£1·00pp

MICRODRIVE COMPATIBLE

Works with QUICKSHOT II & TRACKBALL

Recognised as the only true Hardware Programmed joystick interface this product offers all the features associated with such a design.

You can use *any* Atari-compatible joystick controller with *any* software for your Sinclair Spectrum or ZX81, not just those with a joystick option.

Movement of the joystick is recognised by the computer *exactly* the same as pressing the appropriate control keys, and can therefore give the most immediate response to that movement. The hardware programmed design works with *all* possible key-reading methods, both BASIC and Machine Code.

Eight directional movement, with or without the fire button being pressed, can be achieved by only programming the left, right, up, down and fire keys required by the game.

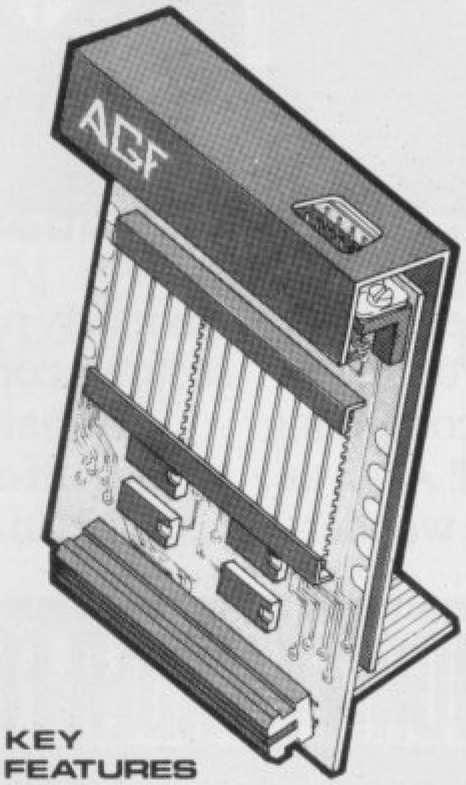
Programming is achieved by a two-digit code, which is looked up on the Programming Chart supplied, for each direction and firing button. These two numbers are then selected on a pair of leads which are clipped onto appropriately numbered strips on the interface.

Once configured this can be marked onto a Quick Reference Programming Card for storing with the game. As the programming is *not* power dependent the interface can be immediately used when next switched on.

The keyboard remains fully functional and can be used simultaneously with the joystick.

An integral rear expansion connector means there is no need to remove the interface to connect other peripherals.

NB. A recent design improvement now means that the AGF Programmable Interface works with the new Quickshot II rapid "Auto Fire" feature.

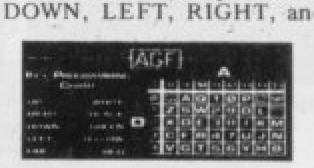


KEY FEATURES

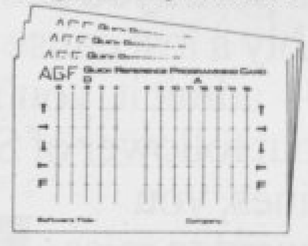
- ★ Programmable design gives TOTAL software support.
- ★ Accepts Atari, Competition Pro, Wico, Starfighter, Quick Shot, Le Stick etc.
- ★ Rear extension connector for all other add-ons.
- ★ Free demo program and instructions.

PACK CONTENTS SUPPLIED

- Programmable Interface Module as illustrated, complete with clip-on programming leads.
- Self adhesive programming chart detailing how to define which key is simulated by UP, DOWN, LEFT, RIGHT, and FIRE.



- One pack of ten Quick Reference Programming Cards for at-a-glance setting to your games requirements.



- 12 months guarantee and full written instructions.

ONLY **11.95** +£1 P&P



Quickshot II® JOYSTICK

NEW IMPROVED GRIP : BUILT-IN STABILIZING SUCTION CUPS

TRIGGER FIRE BUTTON : RAPID AUTO FIRE SWITCH : TOP FIRE BUTTON

AGF Hardware, Bognor Regis, West Sussex PO22 9BY. Telephone: (0243) 823337.

FROM: MR/MRS/MISS — **Please allow up to 28 days for delivery.**

ADDRESS

SEND C.W.O. (NO STAMP NEEDED) TO: A.G.F. HARDWARE, DEPT. **YS**,

FREEPOST, BOGNOR REGIS, WEST SUSSEX, PO22 9BR.

QTY	ITEM	ITEM PRICE	TOTAL
	PROGRAMMABLE INTERFACE	27.95	
	JOYSTICK(S)	12.95	
	PACK(S) QUICK REFERENCE CARDS	1.00	
ZX81 ☐	ZX SPECTRUM ☐ Please tick	FINAL TOTAL	

DEALER ENQUIRIES WELCOME *EXPORT PRICES ON APPLICATION*

VARIABLES ON A THEME

Continuing on from last issue, Dilwyn Jones *completes his investigation of the Spectrum ROM's system variables — those useful housekeeping routines which give the Spectrum its character. And if you fancy delving deeper into your machine, try some of the suggestions* Dilwyn *has to offer. . .*

In the last issue of *Your Spectrum*, an investigation was made of the system variables, 23552 through to 23634. Here we continue with a study of the remaining system variables, 23635 to 23689.

For the uninitiated, the system variables are bytes in the Spectrum ROM which tell the computer the things it needs to know to act in the way you've come to know and love. Information, such as how the memory map is laid out, is held in the system variables so that the computer can access it and update it as and when required.

23635/6 PROG

The address of the start of the area in memory where the Basic program is stored. This points to the first byte of the line number of the first program line. This may be useful if you're converting programs for other Sinclair Research computers with information held as a REM statement in the first line of a program. See also under VARS above.

If you wish to 'security-lock' a line into a program, then by means of this system variable you could POKE a zero into both bytes of a line number at the start of a program. Program lines start with a two-byte line number.

23637/8 NXTLIN

The address of the start of the next program line. You could use this to enable you to access machine code stored within REM statements anywhere in the program, eg. those loaded with MERGE from a tape library of subroutines. These would have their own local calls to machine code like this:

```
9000 LET A=USR (PEEK 23637+256
     *PEEK 23638+5)
9010 REM <> MACHINE CODE <>
9020 RETURN
```

One constraint to this is that you should not include any colour, flash, brightness, etc, control characters into the REM statement or they may be interpreted as machine code, upsetting things somewhat. However, if used from a library of subroutines, these would not normally be used anyway.

23639/40 DATADD

This contains the address of the comma ending the last item of data. If nothing was read from the list (eg. after RUN or restored, etc) the address held in 23639/40 is the address of the byte before the program area, normally the CHR$ 128 at the end of the channel information area. To demonstrate try RUNing this program:

```
10 DATA "1", "2", "3", "4", "5"
20 LET A= PEEK 23639+256*PEEK
   23640
30 PRINT A; TAB 9; PEEK A; TAB 18;
   CHR$ PEEK A AND PEEK A>31
40 READ B$
50 GO TO 20
```

```
23754  128
23763  44   ,
23767  44   ,
23771  44   ,
23775  44   ,
23779  13
```

The address in this two-byte system variable can point to the Enter character or the colon signifying the end of the line or statement containing the data — the address of the terminator of the last item of data.

23641/2 E LINE

This system variable points to the start of the area above the variables. From this we can gain an idea of how much memory is used in bytes by screen, system variables, program and variables, once the program has been RUN to set up the variables, etc. Type this in, as a direct command:

```
PRINT PEEK 23641+256*PEEK
23642-16384
```

We can also tell how much room is used for variables once the program has been RUN to set up the variables. Use the command:

```
PRINT PEEK 23641+256*PEEK
23642-PEEK 23627-256*PEEK
23628
```

23653/4 STKEND

This system variable contains the address of where the spare part of memory starts. From reading this we can gain an idea of how much memory we have left by subtracting it from RAMtop. This will not include memory used for the machine/GOSUB stacks but includes the length of the PEEK statement. So, this is only a fairly accurate guide . . . but one which is adequate from most circumstances.

```
PRINT PEEK 23730+256*PEEK
23731-PEEK 23653-256*PEEK 23654
```

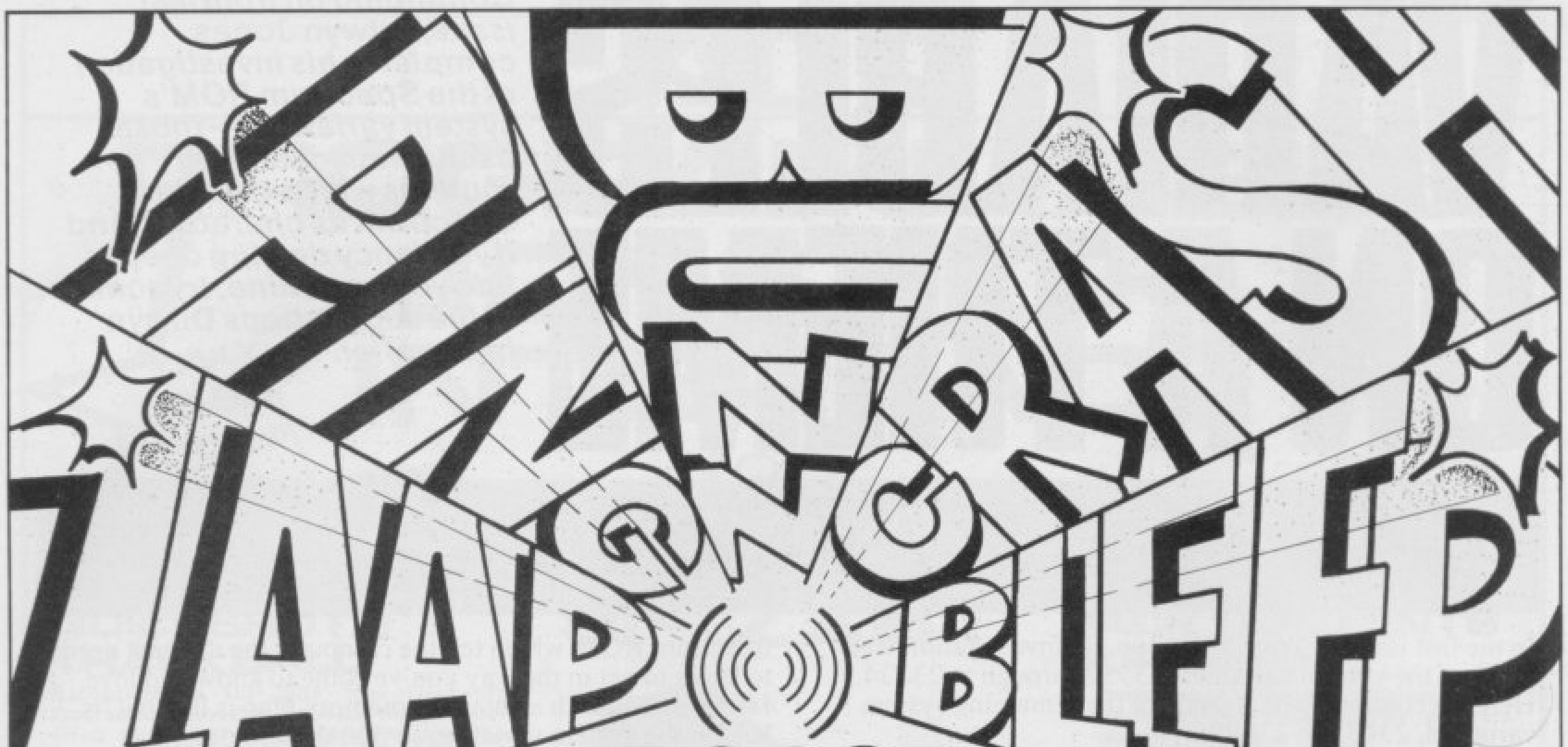
23658 FLAGS 2

This system variable contains some flags used (normally) by the computer to indicate certain conditions.

The best use we can make of this is to utilise the flag indicated by bit 3. This being a one indicates Caps Lock on or Caps Lock engaged.

What use is that? Consider in a program using INKEY$; eg. in a menu of options in a filing program, we often need to know whether the operator is pressing a certain key. If the operator is invited to press 'Y' for Yes or 'N' for No, he/she may press 'y' for Yes or 'n' for No — mixing up lower case and upper case capitals. Most often this would depend on whether Caps Lock was engaged — people are not interested in upper or lower case and whether they press 'y' or 'Y' they expect the computer to understand as humans would. But the computer doesn't really appreciate that. So if we engage Caps Lock automatically, our worries are over and we have a simpler program which doesn't have to check (as far as it's concerned) two separate options for each choice.

It is tempting to use the Basic state-


ZAP
BLEEP

VARIABLES ON A THEME

ment POKE 23658,8 to engage Caps Lock and POKE 23658,0 to disengage it. But this will affect the other flags, so do check their state first unless you know they are not any particular value. Normally in L mode, 23658 has a value of zero so it is generally OK to use the POKEs above. You are not likely to cause crashes, but some funny effects may be caused in rare cases. When the printer buffer is empty, bit 1 will be zero.

23659 DF SZ

This system variable contains the number of lines in the lower section of the screen, normally used for INPUTs, error reports and so on. Normally this would be a two, except for when a long INPUT prompt is displayed, etc. If a value of zero is POKEd in, normally to attempt to clear this unused part so that we can use the whole 24 lines of the screen, the computer crashes.

However, this can be done within a few restrictions. These are that we must ensure the lower part of the screen is restored to normal before any use is made of this — so to break out of a program would be somewhat catastrophic! Also, errors generated within the course of a program will have the same effect since the error report would have to be printed out.

Here is a short listing to demonstrate the use of line 22 and 23 on the screen. Unfortunately, it only works for PRINT or PRINT TAB as we cannot use PLOT down here and PRINT AT will only work down to line 22. The screen is restored to normal by POKE 23659,2 *within* the program.

```
10 POKE 23659,0
20 FOR A=0 TO 23
30 PRINT A
40 NEXT A
50 PAUSE 0
60 POKE 23659,2
```

To demonstrate what can go wrong, let us generate an error by adding this line to the program:

```
45 PRINT error
```

Oops!!! If you just want to PRINT on the bottom two lines it is usually better to use PRINT #1; "text" which works just as well if not better, without such a risk of causing a system crash. If you POKE a value greater than two into DF SZ the upper screen will become smaller than normal. So after POKE 23569, Y the upper screen would be 24-Y rows down and would scroll when the PRINT position got to or beyond 24-Y,0.

This program shows how a part screen scroll can be maintained with DF SZ and SCR CT. Here, random numbers appear and scroll up the top 14 lines of the screen only.

```
10 POKE 23692,0: POKE 23659,10
20 PRINT RND
30 GO TO 10
```

23670/1 SEED

When RANDOMIZE (number) is used, the number (a constant or a variable) is stored in this system variable. This is the number that determines the next random number. It opens up the possibility of cheating, since you could work out the next (supposedly) random number generated and use the knowledge gained to 'swing' luck your way. For example, after RANDOMIZE 1, the next value of RND would be (0.0022735596, INT (RND * 6) + 1) to simulate a die being thrown as a one.

23672/3/4 FRAMES

This is a frame counter which can be used as a timer. It counts frames of a TV picture and so is incremented fifty times a second in the UK, or every 0.02 seconds, although the time taken to actually read and evaluate these three bytes of the timer may not allow it to be used to this accuracy. It has a timing range of nearly four days (actually about three days 21¾ hours). The manual (chapter 18) points out that you need to read the value of these three bytes twice in succession and take the high value for full accuracy because of the possibility of the values of the three changing while being read in such a way as to cause large inaccuracies.

It must be emphasised that the timer bytes are in the opposite order to what you might expect — the most significant byte is 23674, so the timer values are read by:

```
65636*PEEK 23674+256*PEEK
23673+ PEEK 23672
```

which returns time in units of fiftieths of a second.

There are several things that affect the accuracy of this timer. Using BEEP stops the timer. Using the printer and loading/saving, etc, also affect its accuracy. However, the use of PAUSE is OK as this only waits a specified time without re-setting or stopping the timer.

23675/6 UDG

The address of the start of the dot patterns for the user-defined graphics is normally 32600 on a 16K Spectrum or 65368 on a 48K Spectrum. This number is the same as USR "a", so PRINT USR "a" corresponds to:

```
PRINT PEEK 23675+256*PEEK 23676
```

Compulsive POKEers can have fun with this one. The manual suggests changing this to save space by having fewer user-defined graphics. However, it is also possible to do the reverse, and set up more than one user-defined graphics set if required; however, only one set of 21 can be in use at any one time. Remember that since there are 21 UDGs it is necessary to set aside 21 * 8 (168) bytes for each separate set of UDGs and POKE the start addresses, into 23675/6, of the character set in use.

For fun, type in the following commands:

```
POKE 23675,96: POKE 23676,127
(16K Spectrum)
POKE 23675,96: POKE 23676,255
(48K Spectrum)
```

Then, using the user-defined graphics (they normally appear as capital letters until re-defined) try to type out a message. I'll leave you to find out what happens.

One useful tip: once you've set up a user-defined character set it may be SAVEd on tape. Most people would type something like:

```
SAVE "chars" CODE 32600,168
```

Fine, but you have to specify the start addresses. You could use SAVE "chars" CODE(PEEK 23675 + 256 * PEEK 23676), 168, and then you could happily save the current set of UDGs on tape without knowing the address of where they start. This would, for example, allow you to LOAD a character set SAVEd from a 16K Spectrum back into a 48K Spectrum without having to know the addresses.

To get a character set back into the right place on a machine with different amounts of memory, simply use LOAD "chars" CODE(PEEK 23675 + 256 * PEEK 23676),168. This would automatically re-locate data to the right address for the machine in use at the time. This is the same as LOAD "chars" CODE USR "a", which saves a bit of typing although it may look a bit strange.

23679 P POSN

Contains information about how far across the printer buffer the LPRINT position has got to. Contains (33– column number) for columns 0 to 31. You cannot change the LPRINT position by POKEing this alone.

VARIABLES ON A THEME

23680 PRCC

Contains the low byte of the address where the next character is to go into the printer buffer, ie. this will contain (23296 + LPRINT column number), being zero for the left column of the printer, 15 for the 15th column, etc. Because this is the address of the top row of dots of each character, you can POKE this to change the LPRINT buffer position provided you change the value in P POSN (23679) to match. It may appear to work if you don't do this, but problems will be encountered at the end of the line.

23681 Unused system variable

This system variable, although strictly speaking unused, usually contains 91. This is the high byte of the LPRINT buffer address (91 * 256 is 23296 where the buffer starts). This can be POKEd for your own use but using the printer will overwrite it back again to 91. 23680/1 together contain the address of the LPRINT position in the printer buffer. You will not affect the working of the printer if you POKE 23681, but anything stored here may be over-written by the printer routines.

23677/8 COORDS

23677 is the system variable that contains the x co-odinate of the last plotted point. After CLS it starts off at zero and 23678 is the system variable that contains the y co-ordinate of the last point plotted. It contains the actual value, so if the last point plotted was 3,3 both bytes would contain threes.

These two can be POKEd with valid x and y co-ordinates respectively. Since POKEing these does not actually PLOT anything on the screen, this is a convenient way to move the PLOT cursor around. This could be done by PLOT OVER 1;X,Y:PLOT OVER 1;X,Y — but would be messy. Amongst other things this could simulate MOVE found in other BASICs — useful if you wanted to draw lines from around a particular point.

23684/5 DF CC

The address in the display file of the PRINT position. It may be POKEd to send the PRINT output elsewhere, although this requires an understanding of the way the display file is organised.

23688/9 SPOSN

23688 contains information concerning how far across the screen the PRINT position has got. It starts off as 33 for the left-hand side of the screen and decreases by one every time the PRINT position moves one place to the right. After using PRINT AT Y,X; (if Y and X are valid PRINT AT co-ordinates) 23688 would contain 33-X. This can be useful when trying to prevent words being chopped in half when printed on the screen. If you imagine the number in 23688 as counting down towards zero as there is no more room on the current line, you can see that comparing this to the length of the word to be printed gives us an idea of whether it is necessary to move to a new line to prevent the word being chopped. Suppose the word to be printed was W$:

```
IF PEEK 23688 <LEN W$+1 THEN
PRINT
```

This only works for words less than 32 characters long.

23689 contains information relating to how far down the screen the print position has got to. It starts off at 24 for the top line of the screen and decreases by one every time the PRINT position moves down the screen. If you do not want a scrolling display and would rather the screen was cleared when the PRINT position got near the bottom of the screen, then try:

```
IF PEEK 23689=3 THEN CLS
```

23692 SCR CT

Contains the data for how many scrolls will be carried out plus one before waiting with the message 'scroll?' to give viewing time. In graphics games especially this can be a nuisance, since one is not interested in waiting for viewing. So, if the number in 23692 is anything other than one, the waiting does not occur. So POKE 23692,255 would give you 255 lines of printing before the machine waits with 'scroll?'.

POKE 23692,0 seems to have a similar effect except that you have one more line of print. If more is needed, then if the printing is done within a loop, it will be necessary to include the POKE statement in the loop as well — time-wasting but necessary.

23693 ATTR 0

Contains permanent attributes, or the attributes (FLASH, BRIGHT, PAPER and INK) in effect *globally*. Local colours in PRINT statements, etc, are dealt with elsewhere. Note that most of the ROM routines use the values of the system variable holding the *temporary attributes* as these contain the permanent attributes unless a local parameter is specified. CLS, however, clears the screen to the colours, etc, in ATTR P. The functions of the individual bits are shown in *Table 2*.

Bit 7 is one for FLASH 1.

Bit 7 is zero for FLASH 0.

Bit 6 is one for BRIGHT 1.

Bit 6 is zero for BRIGHT 0.

Bits 5, 4 and 3 contain the PAPER colour in binary, eg. for PAPER 7, bits 5, 4 and 3 would be 1, 1 and 1 respectively.

Bit 2, 1 and 0 contain the INK colour in binary, eg. for INK 3, bits 2, 1 and 0 would be 0, 1 and 1 respectively.

Attributes of eight or nine are not dealt with here. If the permanent attributes are eight or nine, then those stored in 23693 may not be valid.

13694 MASK P

This is the system variable that helps the Spectrum determine the attributes of anything printed when a parameter of eight is specified. So, if you specified BRIGHT 8 globally, bit 6 of 23694 would be set to one to remind the computer in future that BRIGHT 8 has been specified. So, to determine the colour/flashing/brightness when printing, the computer looks at what's already there and prints the word in that colour, etc. Or if you like, it only overprints the character on the screen and leaves the attributes alone. You can see what each bit does by looking at *Table 3*.

Bit 7 is one when FLASH 8 is in effect.

Bit 6 is one when BRIGHT 8 is in effect.

Bits 5, 4 and 3 are normally all one when PAPER 8 in effect, but see below.

Bits 2, 1 and 0 are normally all 1 when INK 8 in effect, but see below.

When there is more than one bit to consider, as in INK and PAPER, then only the bits set have their attributes bit taken from the screen. This can lead to some unexpected effects. Try this:

```
10 INK 8
20 POKE 23694, BIN 00000011
30 PRINT AT 0,0; INK 5; "5555555"
40 PRINT AT 0,0; "1111"
```

ADS. THE ULTIMATE IN PERIPHERALS FOR THE SPECTRUM, JUPITER ACE & BBC MICROCOMPUTERS

"Highest quality peripherals at the lowest possible prices" — that's the philosophy at Advanced Digital Systems. Our team of highly experienced designers use the very latest techniques to produce adaptable home computer peripherals — at a price you can afford.

- **Boards are designed using CADCAM, giving a neat optimum layout.**
- **Gold plated connectors are used, wherever possible, to ensure good electrical contact.**
- **All products carry a 12 month guarantee.**
- **All units are "Plug in and go".**

1 PRINTER INTERFACE 1

Centronics Interface for the Sinclair Spectrum. £34.50 inc. VAT. Comes complete with approx 1 metre of cable, centronics plug and software.

- Software recognises LLIST and LPRINT enabling Print-outs direct from Basic. Also adaptable COPY routine for graphics printers.

2 PRINTER INTERFACE 2

Centronics Interface for Jupiter Ace. £39.95 including VAT and Software.

Same unit as the Spectrum Interface but with adapter card.

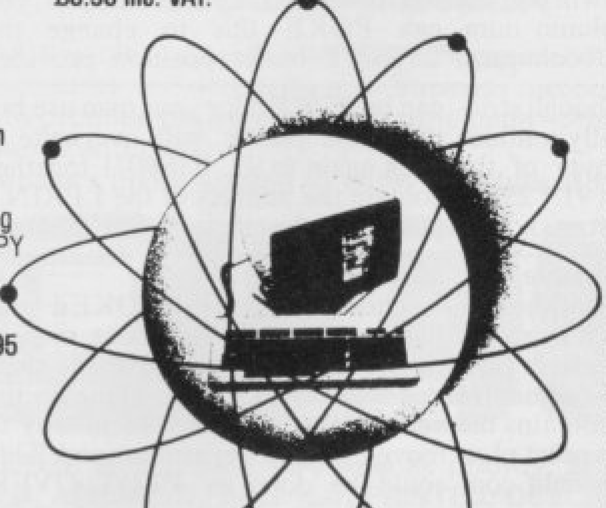
3 PERIPHERAL INTERFACE 3 — ZX ACE

Package Using the adapter card and the Software supplied, the Sinclair ZX Printer can be connected to the Jupiter Ace. Special Introductory Offer £9.95 inc VAT.

4 PERIPHERAL INTERFACE 4

ZX BBC Adapter £24.95 inc. VAT.

Enables the connection of Sinclair Peripherals to the BBC Model B. First Software package included gives an amazingly cheap entry into printing on the BBC model B by allowing you to use the ZX Printer — ideal for listings. Please allow 28 days for delivery. **Unit requires 1.2 Amp power supply: either the ZX P.S.U. or the equivalent available from us at £5.95 inc. VAT.**



NTERSPRINTERSPRINTI

Advanced Digital Systems are now able to supply the following range of printers at amazing prices.

EPSON
- FX 80 — £400
- RX 80 — £278
- RX 80 FT — £306
- MX 80 T/3 — P.O.A.
- MX 100/3 — £453
- FX 100 — £532
- TRACTOR FEED MECHANISM FOR THE FX 80 — P.O.A.

SEIKOSHA
- GP 100 A — £197
- GP 100 VC — £206
- GP 250 X — £246
- GP 700 A (colour printer) — £399
- TEC BUSINESS PRINTERS — P.O.A.

MANNESMAN TALLY
- MT 80 with Parallel Interface £270
- MT 80 with Serial Interface £312

Prices include VAT and delivery

Please send me the items as ticked.

I enclose cheque/order for .... Total
(Include £1.00 for post and packaging).

Cheques payable to Advanced Digital Systems Ltd.

Centronics Base Unit for Sinclair Spectrum £34.50 Inc. VAT & Software ☐
Jupiter Ace Version Base Unit and Adaptor £39.95 Inc. VAT & Software ☐
ZX Ace Adaptor and Software £9.95 Inc. VAT ☐
1.2 Amp P.S.U. £5.95 inc. VAT ☐
ZX BBC Adapter Unit £24.95 Inc. VAT and Software ☐

EPSON
FX 80 — £400 ☐
RX 80 — £278 ☐
RX 80 FT — £306 ☐
MX 100/3 — £453 ☐
FX 100 — £532 ☐

SEIKOSHA
GP 100 A — £197 ☐
GP100 VC — £206 ☐
GP 250 X — £246 ☐
GP 700 A (colour printer) — £399 ☐

MANNESMAN TALLY
MT 80 with Parallel Interface £270 ☐
MT 80 with Serial Interface £312 ☐

PRINTERS — Inc. VAT & Delivery

Name ______________________

Address ______________________

____

Send order to ADVANCED DIGITAL SYSTEMS LTD., 9 Bonchurch Road, Portsmouth, Hampshire PO4 8RY. **YS/2/84**

Advanced Digital Systems Ltd.

MANUFACTURERS AND DESIGNERS OF MICROCOMPUTER PERIPHERALS
9 Bonchurch Road, PORTSMOUTH, Hampshire PO4 8RY. Telephone (0705) 823825.
For International Enquiries (0705) 754160. Prestel Account No. 705-823825



- **Solves storage problems**
- **Computer angled for convenient use**
- **Avoids lengthy setting up**
- **Hides most of wiring**
- **Case and console moulded in black grained ABS plastic**

Treetop Designs, 61 Widmore Road, Bromley. Kent. BR1 3AA.

£44.95 (plus p&p)

Quality executive case, purpose built, with locks, special hinges and lift-off lid with foam lining.

Now available with one of two alternative fitted consoles.

Both allow neat installation of computer, power supply and printer in fitted recesses and incorporate an aluminium switch panel. This eliminates constant removal of plugs for both save/load and power off facilities and provides sockets for tape recorder leads, to allow operation in convenient position shown. Order console S1 for housing tape recorders up to 28cm x 15cm. Order console S2 if your spectrum is fitted with interface 1 and to house up to three microdrives in a fitted recess.

All internal wiring and leads supplied to allow easy plug-in installation.

To: Treetop Designs
Freepost, Bromley, Kent. BR1 3UZ

Please supply S1 ________ cases at £47.45 each (incl. p&p and insurance)
Please supply S2 ________ cases at £47.45 each (incl. p&p and insurance)
My cheque/P.O. for £ ________
is made payable to Treetop Designs

PLEASE PRINT NAME AND ADDRESS

Mr/Mrs/Miss ______________________

Address ______________________

____

____

Y.S.1

Please allow up to 28 days for delivery.

If you are not satisfied we shall of course refund your money, if you return the case undamaged within 14 days.

VARIABLES ON A THEME

As INK 8 is specified, you may expect the ones to be printed in cyan like the fives, but no. Rather than check the INK attribute as a whole, it only checks the bits set in 23694, which were bits 0 and 1. See if you can work out what colour the ones will be printed in. Have fun!

23695 ATTR T

This system variable contains the current temporary colours as would be set up by local statements within PRINT statements. You could see this for your self with something like these two direct commands:

```
PRINT PEEK 23695
PRINT INK 7;PAPER 0; PEEK 23695
```

That is, include the PEEK in a PRINT statement under the effect of the local colour controls. Normally, unless local colour statements are specified, this system variable will contain the global colour values. Colours, etc, to be used for printing on screen are taken from these temporary system variables and things are balanced such that ATTR T is only different from ATTR P if local colour attributes and so on so decree. The function of the individual bits can be seen in *Table 4*.

23696 MASK T

This is rather like MASK P (system variable 23694) except that the parameters here are temporary. Normally the same as the equivalent permanent parameter 8s, this is changed while local colour 8s, etc, are in effect. You could study this by using something like:

```
PRINT PEEK 23696, INK 8; PEEK 23696,
INK 0; FLASH 8; PEEK 23696
```

The individual bits have the functions illustrated in *Table 5*.

23697 P FLAG

This system variable contains, as you might expect from its name, flags used during printing. After PAPER 9 has been specified, bits 6 and 7 are set to one. After INK 9 has been specified, bits 4 and 5 are set to one. After INVERSE 1 has been specified, bits 2 and 3 are set to one. And after OVER 1 has been specified, bits 0 and 1 are set to one. The effects are global if the odd numbered bits (bits 1,3,5 and 7) are set to one, and temporary if the even bits (bits 0,2,4, and 6) are set to 1. See *Table 6* to see what I mean.

23681 23728/9

These three bytes in the system variables are not normally used by the Spectrum — you may like to make use of them as 'custom variables' for use in your own programs in which you need to access information. These are particularly useful in machine code routines where you can simply access the information via an address rather than searching for the variable in the variables area. 23728/9 was intended for use by non-maskable interrupts but these don't occur on the bare Spectrum.

23730/1 RAMTOP

This two byte system variable points to the last byte of RAM of the Basic system area. Note that this is not the end of the memory used by Basic, in the sense that the user-defined graphics normally hide up above this address. If you move RAMTOP up above the start of the user-defined graphics they may be overwritten, but you gain quite a few valuable bytes which may be useful for 16K users.

One important thing is that NEW only operates as far as the address held in 23730/1, so you can store data above this which may be passed between programs loaded into the computer. The same is true if you want to preserve machine code routines, etc.

23732/2 P RAMT

This contains the address of where RAM ends on the Spectrum. If you acquire a Spectrum whose memory capacity you don't know, then don't bother looking inside it to see if it's an expanded model or not, just enter this expression:

```
PRINT PEEK 23732+256*PEEK 23733-16384
```

The 16384 bytes subtracted is for the ROM since RAM starts at address 16384 and goes up to the address held in P RAMT. YS

TABLE 1

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FLASH	BRIGHT	PAPER colour in binary			INK colour in binary		

TABLE 2

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
temp. FLASH	temp. BRIGHT	temp. PAPER colour			temp. INK colour		

TABLE 3

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
temp. FLASH8?	temp. BRIGHT8?	temp. PAPER8?			temp. INK8?		

TABLE 4

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
FLASH8	BRIGHT8	PAPER colour 8?			INK colour 8?		

TABLE 5

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
global PAPER9	temp. PAPER9	global INK9	temp. INK9	global INVERSE1	temp. INVERSE1	global OVER1	temp OVER1

The tables alongside are referenced in the main text of the article, and show the functions of the individual bits of specified system variables: table 1 refers to 23693 ATTR P; table 2 — 23694 MASK P; table 3 — 23695 ATTR T; table 4 — 23696 MASK T; and table 5 — 23697 P FLAG.

This article is extracted from Dilwyn Jones' book, Delving Deeper into your Spectrum ROM *— first published in the UK by Interface Publications, Kensington High Street, London W8 6EJ, and priced at £7.95.*

PLAY POWER!

Whether you're a patient puzzler or an arcade adventurer, there's lots in store in the following pages, courtesy of Colin Young *and* Richard Archdeacon. *And if it's machine code utilities you're after, look no further than the contribution from* Andy Wright.

FLAT CUBE

By Colin Young

This is one of those frustrating puzzles, in much the same vein as Rubic's Cube.

At the start of play, you are presented with a blank 20 by 20 grid. You have the choice of nine levels of difficulty, level one being the easiest and level nine being almost as bad as the original 'cubist' problem. Once you have made your choice, the computer starts to fill in areas of the board — by choosing one square at random, inverting it, and then inverting its eight surrounding neighbours (ie. to the north, east, west and south, and the permutations in-between). Consequently, if any of the blocks the computer is dealing with is already inverted, then the overlapping portion is returned to normal. The number of inversions made is related to the level of play you indicate — which means if you pick level nine, you have one heck of a mess on your hands!

To play the game, you must type in the co-ordinates of the central square of the nine block square you wish to invert. Screen co-ordinates should be typed in with the letter denoting the column first followed by the number of the row. (Don't worry too much about getting it wrong though — it's been thoughtfully error-trapped for the unwary!) If you experiment a little, you will begin to see that, by cautiously re-arranging the grid, the overall screen can be returned to its original blank splendour; of course, you could always play the game the other way round and try to fill the complete grid with black squares, but you would have to re-write a lot of the published listing — not a task for the nervous at heart!

A good tip for playing the game is to try and get all the inverted blocks towards the edge of the screen and then blast them out. Try having a go on the lowest level, just to get an idea of how the game operates, before attempting the more difficult stages. Luck is definitely not the name of this game — but you'll need a good deal of patience!

The idea is to rid the screen of all the black squares. And if you strain the ol' memory back to that popular, if frustrating, cubist nightmare, you may remember that these puzzles are not always as easy as they might at first look!




```
5 POKE 23658,8: REM set caps lock
6 LET goes=0
7 PAPER 1: BORDER 1: INK 7: CLS
8 GO TO 22
```

Lines 5-8 Set the Caps Lock and the 'number of goes' counter, and then initialise the screen attributes.

```
10 CLS : FOR f=0 TO 7: PRINT  PAPER f;
 INK 9;TAB f;"THE FLAT CUBE": NEXT f
20 FOR f=7 TO 0 STEP -1: PRINT  PAPER
f; INK 9;TAB f;"THE FLAT CUBE": NEXT f
21 RETURN
```

Lines 10-21 Contain the subroutine to print the title page.

```
22 GO SUB 10: PRINT AT 20,0;"Do you re
quire instructions y/n."
24 IF INKEY$<>"Y" AND INKEY$<>"N" THEN
GO TO 24
26 IF INKEY$="Y" THEN  GO SUB 6000: RE
M instructions
```

Lines 22-26 Check to see if instructions are required and if so, jump to the 'instructions' routine.

```
30 GO SUB 10: PRINT #1;"DIFFICULTY (1
TO 9) ?"
35 LET d$=INKEY$: IF d$="" THEN  GO TO
35
40 IF d$<"1" OR d$>"9" THEN  GO TO 35
45 LET di=INT VAL d$
50 LET di=di*10
```

Lines 30-50 Draw the title page and prompt the user to input the difficulty level required. The value input, di, is then multiplied by 10, and this new value is the minimum number of goes needed to solve the puzzle.

```
70 GO SUB 1000: REM draw board
80 LET nor=400: LET inv=0
90 GO SUB 3000: REM scramble board
```

Lines 70-90 Draw the blank playing grid, setting the screen flags to show that there are no inverted characters on the screen. The playing area is then set to play, with all the necessary character inversions for the required level.

```
100 REM ----MAIN LOOP-----------
105 LET goes=goes+1
107 PRINT #1;AT 0,0;"Move ";goes
110 INPUT AT 1,1;" Letter & No. "; LINE
m$: IF LEN m$>3 OR LEN M$<2 THEN  GO TO
110
120 LET l$=m$(1): LET m$=m$(2 TO ): IF
l$<"A" OR l$>"T" THEN  GO TO 110
130 IF m$<"1" OR m$>"9" THEN  GO TO 110
140 LET x=CODE l$-63: LET y=VAL m$
150 GO SUB 2000: REM change screen
160 IF NOR=400 OR INV=400 THEN  GO TO 5
000: REM COMPLETED
170 GO TO 100
```

Lines 100-170 Main loop. This area of the program increases the 'number of turns' counter, gets the user's input and checks for its validity. Line 140 converts the user's input to x and y co-ordinates, and line 160 checks to see if the puzzle has been completed.

```
1000 REM ---------Screen-----------
1001 CLS : INK 0
1005 PRINT  PAPER 6;"  ABCDEFGHIJKLMNOPQ
RST  "
1010 FOR f=1 TO 20
1015 IF f<10 THEN  PRINT  PAPER 6;" ";
1020 PRINT  PAPER 6;f; PAPER 7;"++++++++
++++++++++++"; PAPER 6;f;: IF F<10 THEN
 PRINT  PAPER 6;" ";
1025 PRINT
1030 NEXT f
1040 PRINT  PAPER 6;"  ABCDEFGHIJKLMNOPQ
RST  "
1045 INK 7
1050 RETURN
```

Lines 1000-1050 Print the initial blank playing grid.

```
2000 REM -------CHANGE SCREEN---
2020 IF ATTR (y,x)=56 THEN  PRINT AT y,x
; INK 7; PAPER 0;"+": LET inv=inv+1: LET
 nor=nor-1: GO TO 2030
2025 IF ATTR (y,x)<>48 THEN  PRINT AT y,
x; INK 0; PAPER 7;"+": LET nor=nor+1: LE
T inv=inv-1
2030 IF ATTR (y,x+1)=56 THEN  PRINT AT y
,x+1; INK 7; PAPER 0;"+": LET inv=inv+1:
 LET nor=nor-1: GO TO 2040
2035 IF ATTR (y,x+1)<>48 THEN  PRINT AT
y,x+1; INK 0; PAPER 7;"+": LET nor=nor+1
: LET inv=inv-1
2040 IF ATTR (y,x-1)=56 THEN  PRINT AT y
,x-1; INK 7; PAPER 0;"+": LET inv=inv+1:
 LET nor=nor-1: GO TO 2050
2045 IF ATTR (y,x-1)<>48 THEN  PRINT AT
y,x-1; INK 0; PAPER 7;"+": LET nor=nor+1
: LET inv=inv-1
2050 IF ATTR (y+1,x)=56 THEN  PRINT AT y
+1,x; INK 7; PAPER 0;"+": LET inv=inv+1:
 LET nor=nor-1: GO TO 2060
2055 IF ATTR (y+1,x)<>48 THEN  PRINT AT
y+1,x; INK 0; PAPER 7;"+": LET nor=nor+1
: LET inv=inv-1
2060 IF ATTR (y-1,x)=56 THEN  PRINT AT y
-1,x; INK 7; PAPER 0;"+": LET inv=inv+1:
 LET nor=nor-1: GO TO 2070
2065 IF ATTR (y-1,x)<>48 THEN  PRINT AT
y-1,x; INK 0; PAPER 7;"+": LET nor=nor+1
: LET inv=inv-1
2070 IF ATTR (y-1,x-1)=56 THEN  PRINT AT
 y-1,x-1; INK 7; PAPER 0;"+": LET inv=in
v+1: LET nor=nor-1: GO TO 2080
2075 IF ATTR (y-1,x-1)<>48 THEN  PRINT A
T y-1,x-1; INK 0; PAPER 7;"+": LET nor=n
or+1: LET inv=inv-1
2080 IF ATTR (y+1,x-1)=56 THEN  PRINT AT
 y+1,x-1; INK 7; PAPER 0;"+": LET inv=in
v+1: LET nor=nor-1: GO TO 2090
2085 IF ATTR (y+1,x-1)<>48 THEN  PRINT A
T y+1,x-1; INK 0; PAPER 7;"+": LET nor=n
or+1: LET inv=inv-1
2090 IF ATTR (y+1,x+1)=56 THEN  PRINT AT
 y+1,x+1; INK 7; PAPER 0;"+": LET inv=in
v+1: LET nor=nor-1: GO TO 2100
2095 IF ATTR (y+1,x+1)<>48 THEN  PRINT A
T y+1,x+1; INK 0; PAPER 7;"+": LET nor=n
or+1: LET inv=inv-1
2100 IF ATTR (y-1,x+1)=56 THEN  PRINT AT
 y-1,x+1; INK 7; PAPER 0;"+": LET inv=in
v+1: LET nor=nor-1: GO TO 2110
2105 IF ATTR (y-1,x+1)<>48 THEN  PRINT A
T y-1,x+1; INK 0; PAPER 7;"+": LET nor=n
or+1: LET inv=inv-1
```

Lines 2000-2105 This long routine inverts the screen surrounding and including the position you have chosen (x,y). Each square is inverted individually. (This routine is not the best-written we have seen — in fact, it might be well worth your time employing a loop to try and shorten it.)

```
2110 RETURN
3000 REM -----SCRAMBLE----------
3010 FOR f=1 TO di
3020 LET x=INT (RND*20)+2: LET y=INT (RN
D*20)+1
3030 GO SUB 2000
3040 NEXT f
3050 RETURN
```

Lines 2110-3050 Contain the 'scramble' routine, used by the computer at the start of each game to invert the nine-square blocks di times (di being the value you input for the level of difficulty).

```
5000 REM ------COMPLETED---------
5010 FOR f=0 TO 30 STEP .3
5020 BEEP .003,f: BEEP .003,f+10
5030 NEXT f
5040 CLS : PRINT AT 10,3; INK 7; PAPER 5
; FLASH 1; BRIGHT 1;"YOU'VE DONE IT BRAI
NY": PRINT AT 15,0;"DO YOU WANT TO HAVE
ANOTHER GO?"
5045 PRINT AT 12,4;"IT TOOK YOU "; FLASH
 1;goes; FLASH 0;" goes"
5050 IF INKEY$="" THEN  GO TO 5050
5060 IF INKEY$="Y" THEN  GO TO 1
5070 STOP
```

Lines 5000-5070 When the puzzle is completed, the 'completion' routine tells you how many 'goes' it took you and offers you another game. You will also get a message of congratulations — which on level nine, does your ego no end of good!

```
5999 REM instructions
6000 CLS
6010 PRINT TAB 5;"*********************"
6020 PRINT TAB 5;"*    THE FLAT CUBE    *"
6030 PRINT TAB 5;"*********************"
'''
6040 PRINT "Flat Cube is a 2D Rubic's Cu
be. The board has a 20x20 grid of   plus
 signs. The Spectrum will   select one o
f these at random & invert it and its 8
neighbours. This process is repeated a
    number of times according to thediff
iculty level."
6050 PRINT ''"Your simple";
6060 FOR i=1 TO 6: PRINT CHR$ 8;: NEXT i
6070 PRINT  OVER 1;"______"; OVER 0;" ta
sk is to return   the board to its origi
nal state i.e. blank."
6080 PRINT AT 20,5; FLASH 1;"PRESS ANY K
EY TO PLAY"
6090 IF INKEY$="" THEN  BEEP .1,10: GO T
O 6090
```

```
6100 FOR i=1 TO 20: BEEP .1,i: NEXT i
6110 RETURN
```

Lines 5999-6110 Print out the instructions should you have requested them (see lines 22-26).

REM REMOVER

By Andy Wright

If you've written a routine following all the rules cited in the text books, you'll more than likely end up with a listing full of REM statements that have so faithfully guided you throughout its inception. Unfortunately, you may find that once completed and working correctly, the REM statements take up valuable memory you require for the main program.

This program for the 16K or 48K Spectrum provides the very useful facility of REM removal. First, type in the program as listed, taking extra care with the data held in lines 1000 to 1230. Now save the program to tape and then, once verified, RUN it. A machine code call address will be displayed on the screen, along with the message 'Save Code to Cassette Y/N?'. Press the 'n' key and then type in the USR call as displayed (do remember to make a note of the call address *before* you press 'y' or 'n').

If all has gone well, the 'OK' report should be given and once listed, you will find all the REM statements and the comments within them have been removed. If this hasn't happened, not to worry — simply reset the computer, load the program back from tape, carefully re-check all the data statements and try again.

If the REM statements have indeed vanished, type RAND USR 0 to reset the computer and load the program back from tape again. This time, when confronted with the 'Save' question, press the 'y' key and save the machine code to tape. (Don't forget to make a note of the call address given on-screen, preferably on the cassette itself.)

Before you rush off and type this into your computer in a burst of enthusiasm, there are one or two things you should know which could be helpful. First off, if the Basic program you are developing changes the values of any system variables, such as UDG or RAMTOP (ie. by executing a CLEAR command), it may corrupt the utility. So, use the utility *before* running such a program and re-load the utility before using it again.

You might also like to know that you can locate this program anywhere in memory you choose; thus, you can overcome a problem such as the one described above. You should first save the machine code version of the utility as described above, but load and run it using commands of the following format:

```
CLEAR xxxxx-1
LOAD "rema" CODE xxxxx
```

where 'xxxxx' is a suitable area in memory (and 'xxxxx-1' is one less than this number).

Then, to remove the REMs from a program, simply type:

```
PRINT USR xxxxx
```

You should also note that if the only statement on a line is a REM statement then the whole line is deleted from the program. Also, if the REM is the last statement on a line then only the REM part of the line is deleted.

One last note is that this utility should only be used once you are sure you won't need the REM statements for some future reference. So, either be quite certain you don't need the information they contain, or make a hard copy first — just in case!

```
  10 GO SUB 500:  REM work  out memory
 location  to hold first byte of machine
 code
  20 CLEAR progsta: REM protect the  are
a  of memory to receive the machine code
  25 PRINT "Please wait while code poked
."
  30 GO SUB 500: GO SUB 700: REM   work
out memory locations to receive  start
&  end bytes of machine code
```

Lines 10-30 Set up an area for the machine code to reside.

```
  40 FOR i=progsta   TO progend: REM  lo
op  to poke machine code into protected
memory
  50    READ byte
  60    POKE i,byte
  70 NEXT i
```

Lines 40-70 Read in and POKE the machine code.

```
  80 CLS
  90 PRINT "to remove REMs use the comma
nd"
 100 PRINT : PRINT "PRINT USR ";progsta
 110 PRINT ''"Save Code To Cassette Y/N?
"
 120 IF INKEY$="y" OR INKEY$="Y" THEN  G
O TO 200
 130 IF INKEY$="n" OR INKEY$="N" THEN  S
TOP
 140 GO TO 120
```

Lines 80-140 Print the address of the machine code and ask if a SAVE of the code is required.

```
 200 CLS : PRINT ''"Remove EAR lead."
 210 SAVE "Rem Remove"CODE progsta,128
 220 CLS : PRINT ''"Rem remover code now
 saved."''"To use the utility at any tim
e  type the following:"
 230 PRINT '"CLEAR ";progsta-1;'"LOAD ""
 CODE"'"PRINT USR ";progsta
 240 PRINT ''"Write down the above detai
ls nowfor later use.": STOP
```

Lines 200-240 The 'cassette SAVE' routine for the machine code.

```
 500 LET ramtop=23730: REM pointer to en
d of basic
 520 LET proglen=140: REM number of byte
 of memory to be protected
 530 LET progsta=(256*PEEK (ramtop+1)+PE
EK ramtop)-proglen: REM memory location
 to receive first byte of machine code
 540 RETURN
```

Lines 500-540 The subroutine to work out the position in memory the machine code will be stored.

```
 700 LET progsta=progsta+proglen: REM fi
rst byte to receive machine code - allow
ing for changes in values of system vari
ables caused by CLEAR statement in line
20
 710 LET progend=progsta+127: REM memory
 location to receive last byte of machin
e code program
 720 RETURN
```

Lines 700-720 Calculate the start and end positions of the machine-code.

```
1000 DATA 42,83,92,237,91,75
1010 DATA 92,229,167,237,82
1020 DATA 225,32,6,201,179,30
1030 DATA 195,169,18,229,35,35
1040 DATA 229,35,6,0,35,126,2' 1
1050 DATA 14,32,7,35,35,35,35
1060 DATA 35,24,243,254,13,32
1070 DATA 4,209,209,24,77
1080 DATA 254,234,40,10,254,58
1090 DATA 32,227,84,93,6,1,24
```

```
1100 DATA 221,213,229
1110 DATA 209,35,126
1120 DATA 254,13,32,250
1130 DATA 209,175,184,32,10,209
1140 DATA 209,35,213,205,229,25
1150 DATA 225,24,173,34,251
1160 DATA 255,237,83,253
1170 DATA 255,167,237,82,209
1180 DATA 213,235,78,35,70,197
1190 DATA 225,167,237,82,235
1200 DATA 225,115,35,114,225
1210 DATA 42,251,255,237,91,253
1220 DATA 255,213,205,229,25
1230 DATA 225,35,24,131
```

Lines 1000-1230 Contain the DATA for the machine code.

HOPALONG

By Richard Archdeacon

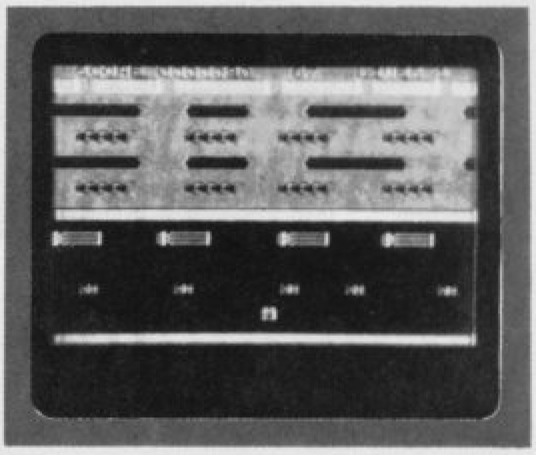
Here's a new version of an old arcade classic — namely *Frogger*. Written for the 48K Spectrum, the program contains three levels, and instructions are provided within the annotations for you to construct a fourth should you feel adventurous.

In much the same way as the arcade original, your task is to get the frog character from the bottom of the screen, negotiating the various hazards and then leaping into one of the four available spaces at the top of the screen. However, you'll soon find that things are not quite that easy.

Your problems begin when you try and cross the road — there's a fair amount of traffic to hop around. Needless to say, one hop in the wrong direction and it's frog's legs on the menu tonight! Once across the road, you must get across the river via the floating logs and turtles. And if you think all that sounds well within your grasp, you may be interested to know that all four of your frogs must be hopped home against the clock.

The program follows the original fairly faithfully, so there's a good chance most of you will know the rudiments of the game. Movement of your frog character is achieved with the cursor control keys. And at the end of the game, a table is displayed giving the top five scores of the day.

The speed of the game is fast for Basic, and compares well with many commerical packages. The program is generally well-structured, although the more experienced programmers amongst you may want to have a closer look at the user-defined graphics subroutine.



As you can see, the game reproduces all the horrors of guiding your frog character across the road and river. Be careful though, the Green Cross Code Man won't be there to help you this time!

```
  1 REM © R.Archdeacon
  2 REM Annotated by Gavin Monk
 10 DIM h(6): GO SUB 9000: REM START
 20 LET s=0: LET li=3:
```

Lines 10-20 These two lines initialise the program. (s=score, li=lives remaining and h() is the high score array.)

```
  30 GO SUB 7500: REM LEVEL
  35 LET fr=0
  40 GO SUB 7000: REM SCREEN
  50 LET t=100: LET a=21: LET b=16: PRIN
T AT a,b; INK 4;"e"
```

Lines 30-50 New level initialisation. (fr=the frog's home, t=the time left, and a and b are the frog's co-ordinates.)

```
  60 REM MAIN LOOP
  65 BEEP .005,-15
  66 LET t=t-1: PRINT AT 0,18; INK 0; PA
PER 5;t;"  "
  67 IF t<=0 THEN  GO TO 3000
```

Lines 60-67 Start of the main loop. This reduces the time (t) by one and prints the current value on the screen. Line 67 checks to see if you have run out of time.

PROGRAMMING ON SPEC

Programs submitted for publication in *Your Spectrum* have a tendency to suffer a number of common faults — simple little quirks which can be rectified easily.

Our major gripe is that the display is often messy and untidy. A screen can be made so much more legible if it is well-formatted — if spaces are left between lines of text and words are not split from line to line. And while we're on about screens, if you display something like the current score or whatever, do remember that labelling it makes all the difference — the whole effect becomes more user-friendly (to use a rather hackneyed phrase). Other suggestions for improvements are:

1. Use the 'PRINT #1;' statement to allow you to print on the bottom two lines of the screen.
2. Use INKEY$ for simple inputs which only require a single key response.
3. Check that answers to any questions within a program are valid.
4. If long calculations are carried out within the program's operation don't let the user sit there idly twiddling thumbs, print a message such as 'please wait — I'm thinking'. Or even better, give the user something to read, such as the instructions for playing the game, while the calculation is being undertaken.
5. Write major sections of your program as subroutines, and label them with REM statements explaining what they all do. It is always useful to place a REM statement after a GO SUB call, for instance.
6. If your program is written in machine code, or contains a machine code routine, then provide a checksum for the data so that the unwary typist will realise their mistake before it's too late. An Assembly listing accompanying the code would also be useful.
7. Write your listings in as structured a form as you can manage, as this will help other programmers to improve and/or modify them.

Above all, think of all the annoying aspects of listings you've experienced in your programming career — and make sure they don't happen in your programs. That said, the standard of contributions to *Your Spectrum* have been extremely high. So, if you want to see your name *YS*, then we want to see your programs — remember, we pay better than most, but they've got to be good!

Please send any programming contributions to:
Gavin Monk, Your Spectrum, 14 Rathbone Place, London W1P 1DE.

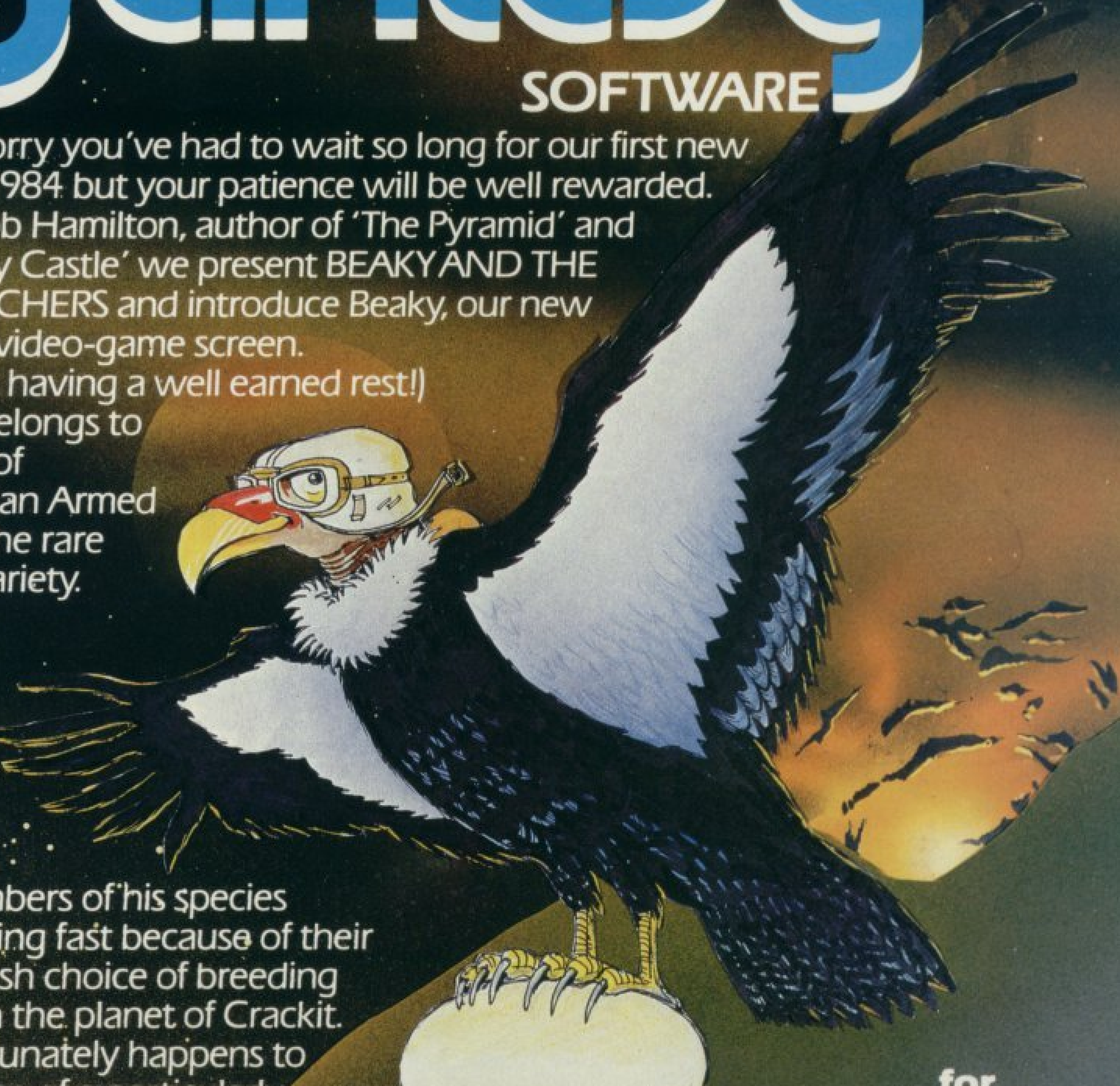
fantasy SOFTWARE

We're sorry you've had to wait so long for our first new release of 1984 but your patience will be well rewarded.

From Bob Hamilton, author of 'The Pyramid' and 'Doomsday Castle' we present BEAKY AND THE EGGSNATCHERS and introduce Beaky, our new star of the video-game screen.

(Ziggy is having a well earned rest!)

Beaky belongs to the family of Andromedan Armed Condors, the rare goggled variety.



The numbers of his species are dwindling fast because of their rather foolish choice of breeding ground, on the planet of Crackit. This unfortunately happens to be the home of a particularly evil band of creatures known collectively as the Eggsnatchers. Their sole malicious intent in life is to extinctify Beaky's species by stealing or destroying the eggs by any foul means available.

BEAKY and the Egg Snatchers

for Spectrum 48K and Commodore 64

Your objective is to try and brood, hatch out and rear as many little Beakys as possible through 12 different screens of formidable but delightful arcade action.

FANTASY SOFTWARE is available from W.H.SMITHS, JOHN MENZIES, BOOTS, WOOLWORTHS, LASKYS, GREENS, RUMBELLOWS, SPECTRUM GROUP and all other good software retailers.

Beaky and the Eggsnatchers is available for 48K Spectrum and Commodore 64 at £6.50 on cassette or on disk for the Commodore 64 at £9.50 from

FANTASY SOFTWARE, FAUCONBERG LODGE, 27A ST GEORGES ROAD, CHELTENHAM, GLOS GL50 3DT
despatched by return first class post together with free membership of the Fantasy Micro Club.
Trade Enquiries welcome – telephone 0242-583661.


"GET AWAY FROM IT ALL"
NEW!! FREEZ' BEES (48K)
Penguin Pulvirizin!
NEW!! SAM SPADE
The Fastest Digger around.
NEW!! ROBOT RIOT (48K)
Metal Mayhem at the
ACME ROBOT CO.
MISSION IMPOSSIBLE
It certainly IS.
EXTERMINATOR
Last of the ROBOTRONS
ARMAGEDDON
Get your Apocolypse NOW!
BRAIN DAMAGE
Crashed out CPU.
WITH OUT OF THIS WORLD
SPECTRUM GAMES
Selected titles available from Boots, W.H.Smiths, J.Menzies and all good computer stores.
Dealer Enquiries welcome Tel: 01-748-4125
All games just £5.95 inc VAT and P&P
Silversoft Ltd, London House, 271/273 King St, London W6
SILVER SOFT

```
 70 LET a$=a$(32)+a$( TO 31): LET h$=h$
(32)+h$( TO 31)
 80 LET b$=b$(2 TO )+b$(1): LET e$=e$(2
TO )+e$(1): LET g$=g$(2 TO )+g$(1)
 90 PRINT  PAPER 5;AT 3,0; INK 0;a$;AT
5,0; INK 2;b$;AT 7,0; INK 0;a$;AT 9,0; I
NK 2;b$
 95 BEEP .005,-15
 100 PRINT  PAPER 0;AT 13,0; INK 7;e$;AT
 15,0; INK 2;h$;AT 17,0; INK 3;g$;AT 19,
0; INK 2;h$
 110 IF a=3 OR a=7 THEN  LET b=b+1: IF b
=31 THEN  GO TO 3000
 115 IF a=5 OR a=9 THEN  LET b=b-1: IF b
=0 THEN  GO TO 3000
 120 GO SUB 200
```

Lines 70-120 The 'main movement' routine. Lines 70 and 80 rearrange the strings so they look like they are moving from left to right or right to left. Lines 90 and 100 print the objects in their new positions. Lines 110-115 check to see if the frog is on a turtle or log, and move the frog left or right as required. Also, these lines check to see if the frog has hit the edge of the screen.

```
 130 IF INKEY$="7" THEN  GO TO 1000
 140 IF INKEY$="8" OR INKEY$="5" THEN  G
O SUB 300
 150 IF a=3 OR a=7 OR (a>11 AND a<21) TH
EN  PAPER 0
 151 IF a=5 OR a=9 THEN  PAPER 2
 152 IF a=11 OR a=21 THEN  PAPER 1
 153 PRINT AT a,b; INK 4;"e"
 160 IF INKEY$="6" THEN  GO SUB 1500
 180 GO TO 60
```

Lines 130-180 The 'main key scan' routine. This routine checks to see if you've pressed one of the cursor keys and if so, jumps to left, right, up or down movement. The PAPER colour is set depending on the frog's position.

```
 200 IF a=21 OR a=11 THEN  RETURN
 201 GO TO 200+a
 202 RETURN
 203 IF a$(b+1)=" " THEN  GO TO 3000
 204 GO TO 202
 205 IF b$(b+1)=" " THEN  GO TO 3000
 206 GO TO 202
 207 GO TO 203
 209 GO TO 205
 213 IF e$(b+1)<>" " THEN  GO TO 3000
 214 GO TO 202
 215 IF h$(b+1)<>" " THEN  GO TO 3000
 216 GO TO 202
 217 IF g$(b+1)<>" " THEN  GO TO 3000
 218 RETURN
 219 GO TO 215
```

Lines 200-219 This routine checks to see if the frog has hit a car or lorry, or moved into the water. If so, the action jumps to the 'dead frog' routine.

```
 300 IF a=3 OR a=7 OR (a>11 AND a<21) TH
EN  PAPER 0
 301 IF a=5 OR a=9 THEN  PAPER 2
 302 IF a=11 OR a=21 THEN  PAPER 1
 303 PRINT AT a,b;" "
 310 BEEP .05,10: LET b=b+(INKEY$="8" AN
D b<31)-(INKEY$="5" AND b>0)
 320 PRINT AT a,b; INK 4; PAPER 1; PAPER
 0 AND a>11 AND a<21; PAPER 5 AND a<11;"
e"
 330 RETURN
```

Lines 300-330 This routine checks to see if the frog is still on the screen and increments or decrements b if moved left or right, respectively. The PAPER colour is set depending on the frog's position. Line 320 prints the frog in its new position.

```
1000 IF a=3 OR a=7 OR (a>11 AND a<21) TH
EN  PAPER 0
1001 IF a=5 OR a=9 THEN  PAPER 2
1002 IF a=11 OR a=21 THEN  PAPER 1
1003 PRINT AT a,b;" "
1010 LET a=a-2: BEEP .05,10
1015 IF a=1 THEN  GO TO 2000
1017 GO SUB 200
1050 PRINT AT a,b; INK 4; PAPER 1; PAPER
 0 AND a>11 AND a<21; PAPER 5 AND a<11;"
e"
1060 LET s=s+10: PRINT AT 0,15-LEN STR$
s; PAPER 5; INK 0;s
1090 GO TO 60
```

Lines 1000-1090 The 'frog moved up' routine. This routine moves the frog up the screen. Line 1015 checks to see if the frog is home. Line 1017 calls the 'hit object yes/no' routine. Line 1050 prints the frog in the new position. Line 1060 increments and prints score.

```
1500 IF a=3 OR a=7 OR (a>11 AND a<21) TH
EN  PAPER 0
1501 IF a=5 OR a=9 THEN  PAPER 2
1502 IF a=11 OR a=21 THEN  PAPER 1
1503 PRINT AT a,b;" "
1505 IF a=21 THEN  RETURN
1510 LET a=a+2: BEEP .05,10
1540 GO SUB 200
1550 PRINT AT a,b; INK 4; PAPER 1; PAPER
 0 AND a>11 AND a<21; PAPER 5 AND a<11;"
e"
1590 GO TO 60
```

Lines 1500-1590 The 'frog moved down' routine. As for 'move-up' routine, but there is no check for 'home'.

```
2000 IF b<>2 AND b<>8 AND b<>16 AND b<>2
3 AND b<>29 THEN  GO TO 3000
2001 IF ATTR (1,b)=44 THEN  GO TO 3000
2002 LET s=s+100: PRINT AT 0,15-LEN STR$
 s; PAPER 5; INK 0;s
2003 PRINT AT 1,b; INK 4; PAPER 5;"d"
2005 LET fr=fr+1: BEEP .05,15: BEEP .05,
20
2010 IF fr=5 THEN  GO TO 2200
2020 GO TO 50
```

Lines 2000-2020 The 'check if home' routine. Line 2000 checks to see if the frog is in a possible home position and if not, the action jumps to the 'dead frog' routine. Line 2001 checks to see if all the home positions are full and if not, the score is incremented and printed. Line 2010 checks to see if all the frogs are home.

```
2200 PRINT AT 11,10; INK 7; BRIGHT 1; PA
PER 1;"BONUS ";t*10
2210 FOR f=0 TO 3: BEEP .1,2: BEEP .1,4:
 BEEP .1,6: BEEP .1,6: BEEP .1,4: BEEP .
1,0: NEXT f: LET s=s+1000
2215 LET s=s+(t*10): PRINT AT 0,15-LEN S
TR$ s; PAPER 5; INK 0;s
2220 FOR f=1 TO 5: FOR n=1 TO 5: BEEP .0
1,f+n: NEXT n: NEXT f: GO TO 30
```

Lines 2200-2220 The 'all frogs home' routine. Line 2200 prints the time bonus, and line 2210 plays the tune and increments the score.

INFORMSOFTWARE

FINANCIAL PACKAGES FOR THE SPECTRUM 48K

DO YOU WANT TO KNOW WHAT TYPE OF INSURANCE/ASSURANCE/INVESTMENTS MEETS YOUR OWN PERSONAL REQUIREMENTS?

LET YOUR COMPUTER EARN IT'S KEEP

£x% = £+?

THIS PROGRAM IS DESIGNED TO PROVIDE YOU WITH INFORMATION WHICH WILL LET YOU DISCOVER THE WORLD OF PERSONAL FINANCE AS IT DIRECTLY RELATES TO YOU THE INTENTION IS TO FOLLOW THE PROGRAM THRO' A NUMBER OF TEACHING AND DISPLAY ROUTINES WITH THE FACILITY TO CARRY OUT PERSONAL INVESTMENT CALCULATIONS THE SUBJECTS COVERED INCLUDE WHOLE LIFE, TERM, ENDOWMENT ASSURANCE, HOUSEHOLD/CAR/TRAVEL-HEALTH/PENSIONS/BUSINESS INSURANCE, INSTANT QUOTES ARE AVAILABLE.
THE EXTENSIVE SECTION ON INVESTMENT PROVIDES PRELIMINARY GUIDANCE FOR VARIOUS TYPES OF INVESTMENT INCLUDING WHAT TO DO WITH A WINDFALL ENABLING YOU TO LOOK COMPREHENSIVELY AT ALL THOSE ASPECTS OF YOUR INVESTMENT DECISIONS. THREE INVESTMENT CALCULATORS COVERING COMPOUND INTEREST RETURNS ON A PORTFOLIO OF INVESTMENTS AND THE DIFFICULT DECISION AS TO WHETHER YOU SHOULD COMMUTE YOUR PENSION ARE INCLUDED.
LAST YEARS TAX AND THIS YEARS PROPOSED ALLOWANCES ARE SHOWN THE FACILITY TO HOLD/OR PRINT OUT PAGES IS AVAILABLE. A MICRODRIVE VERSION IS AVAILABLE.

CHEQUES FOR £5.50 TO

INFORMSOFTWARE

**3, TREESDALE CLOSE,
BIRKDALE
SOUTHPORT, PR8 2EL.**

TRADE ENQUIRIES WELCOME

```
3000 REM FROG DEAD
3010 PRINT AT a,b; INK 6; FLASH 1; BRIGH
T 1;"e"
3020 LET li=li-1: PRINT AT 0,23; PAPER 5
; INK 0;"LIVES ";li
3030 IF li<=0 THEN  GO TO 4000
3040 FOR f=30 TO 1 STEP -1: BEEP .05,f:
NEXT f: PRINT AT a,b; INK 3;"■": GO TO 5
0
```

Lines 3000-3040 The 'dead frog' routine. This routine is called after a crash or the allowed time has elapsed. Line 3010 prints a flashing frog, and line 3020 prints the lives left after one has been taken. Line 3030 checks to see if all the lives have gone and if so, the computer plays a tune.

```
4000 REM GAME OVER
4010 BEEP .1,30: PRINT AT 1,11; FLASH 1;
 PAPER 6; INK 0;"GAME OVER"
4020 BEEP 1,-10: BEEP 1,-25: BEEP 1,-40:
 LET lev=1
4030 LET h(6)=s: FOR f=1 TO 6: FOR n=1 T
O 5: LET z=h(n): LET z1=h(n+1): IF z<z1
THEN  LET h(n)=z1: LET h(n+1)=z
4040 NEXT n: NEXT f
4050 PAPER 5: BORDER 6: CLS : PRINT AT 2
,0; INK 2;x$: FOR f=1 TO 5: PRINT AT (f*
2)+5,10;f;" - ";h(f): NEXT f
4060 FOR f=1 TO 5: IF h(f)=s THEN  PRINT
 AT (f*2)+5,10; FLASH 1;f;" - ";h(f): GO
 TO 4080
4070 NEXT f
4080 PRINT AT 20,4;"PRESS ANY KEY TO BEG
IN"
4090 IF INKEY$="" THEN  BEEP .003,RND*3:
 GO TO 4090
4100 BEEP .01,40: GO TO 20
```

Lines 4000-4100 The 'game is over' routine. Line 4020 plays a tune and resets the level back to level one. Lines 4030 and 4040 sort out the high score table. Lines 4050-4080 print the title and high score table, complete with flashing highest score. Line 4090 BEEPs while waiting for a key to be pressed.

```
7000 REM SCREEN
7010 BORDER 6: PAPER 5: CLS  ·
7020 PRINT AT 1,0; INK 3;"■■ ■■■■■ ■■■■■
■■ ■■■■■■ ■■■■■ ■■";AT 11,0; INK 1;"■■■■
■■■■■■■■■■■■■■■■■■■■■■■■■■■■";AT 21,0; I
NK 1;"■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■";
AT 0,2; INK 0;"SCORE 0000000";AT 0,23; I
NK 0;"LIVES ";li
7030 PRINT AT 0,15-LEN STR$ s;s
7040 FOR f=12 TO 20: PRINT AT f,0; INK 0
;"■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■": NEX
T f
7050 PRINT  PAPER 5;AT 3,0; INK 0;a$;AT
5,0; INK 2;b$;AT 7,0; INK 0;a$;AT 9,0; I
NK 2;b$
7060 PRINT  PAPER 0;AT 13,0; INK 7;e$;AT
 15,0; INK 2;h$;AT 17,0; INK 3;g$;AT 19,
0; INK 2;h$
7070 RETURN
```

Lines 7000-7070 Draw the initial screen.

```
7500 REM LEVEL
7510 IF lev>3 THEN  RETURN
7520 GO SUB 7500+(100*lev)
7530 LET lev=lev+1: RETURN
7600 REM LEVEL 1
7610 LET a$=" b■■■■■■■■c     b■■■■■■■■c    b■■
■c    "
7620 LET b$="    aaaa     aaaa      aaaa
 aaaa"
7650 LET e$="    ghhi     ghhi    ghhi     g
hhi   "
7670 LET g$="    1m   1m      1m   1m
1m    "
7680 LET h$="    jk      jk      jk     jk
  jk "
7690 RETURN
7700 REM LEVEL 2
7710 LET a$="           b■■■■■■c         b■■■■
c    "
7720 LET b$="    aaaa      aaaa      aaaa
    "
7750 LET e$="   ghhi    ghhi  ghhi   ghhi
ghhi "
7770 LET g$=" 1m    1m    1m  1m    1m    1m
    "
7780 LET h$=" jk     jk    jk    jk    jk
jk    "
7790 RETURN
7800 REM LEVEL 3
7820 LET b$="           aaaa         aaaa
    "
7850 LET e$="   ghhi    ghhi  ghhi   ghhi
ghhi "
7870 LET g$=" 1m  1m    1m  1m    1m    1m
  1m "
7880 LET h$=" jk    jk    jk    jk  jk    jk
  1m "
7890 RETURN
```

Lines 7500-7890 This routine sets up the number of cars, lorries, logs and turtles which appear in each lane on the screen. The variable 'lev' in line 7520 causes a jump to the required data for a particular level. Should you wish to add another level to the game, you must first change the '3' in line 7510 to a '4'. You must then add the following lines:

```
7900 REM LEVEL 4
7910 LET a$= "      logs       "
7920 LET b$= "      turtles    "
7930 LET e$= "      lorries    "
7940 LET g$= "      cars       "
7950 LET h$= "      racing cars "
7960 RETURN
```

Obviously, you can put as many graphic characters you want in the above strings — but you must make sure that the strings contain 32 characters.

```
8000 REM U.D.G.
8010 FOR f=0 TO 7: READ a: POKE USR "a"+
f,a: NEXT f
8020 DATA BIN 10000011,BIN 0101110,BIN 0
1111110,BIN 11111110,BIN 11111110,BIN 01
111110,BIN 01011100,BIN 10000011
8030 FOR f=0 TO 7: READ a: POKE USR "b"+
f,a: NEXT f
8040 DATA BIN 00011111,BIN 0011111,BIN 0
0111111,BIN 01111111,BIN 01111111,BIN 00
111111,BIN 00111111,BIN 00011111
8050 FOR f=0 TO 7: READ a: POKE USR "c"+
f,a: NEXT f
8060 DATA BIN 11111000,BIN 11111000,BIN
11111100,BIN 11111110,BIN 11111110,BIN 1
1111100,BIN 11111100,BIN 11111000
8070 FOR f=0 TO 7: READ a: POKE USR "d"+
f,a: NEXT f
8080 DATA BIN 11100111,BIN 1011101,BIN 1
1111111,BIN 01111110,BIN 11011011,BIN 11
100111,BIN 01111110,BIN 10000001
8090 FOR f=0 TO 7: READ a: POKE USR "e"+
f,a: NEXT f
8100 DATA BIN 0,BIN 01000010,BIN 0101101
0,BIN 01111110,BIN 00111100,BIN 01111110
,BIN 01111110,BIN 00111100
8130 FOR f=0 TO 7: READ a: POKE USR "g"+
f,a: NEXT f
```

```
8140 DATA BIN 00111100,BIN 01011101,BIN
01011111,BIN 01011101,BIN 01011101,BIN 0
1011111,BIN 01011101,BIN 00111100
8150 FOR f=0 TO 7: READ a: POKE USR "h"+
f,a: NEXT f
8160 DATA BIN 11111111,BIN 11111111,BIN
0,BIN 11111111,BIN 11111111,BIN 0,BIN 11
111111,BIN 11111111
8170 FOR f=0 TO 7: READ a: POKE USR "i"+
f,a: NEXT f
8180 DATA BIN 11111111,BIN 11111111,BIN
00000011,BIN 11111111,BIN 11111111,BIN 0
0000011,BIN 11111111,BIN 11111111
8190 FOR f=0 TO 7: READ a: POKE USR "j"+
f,a: NEXT f
8200 DATA BIN 00101110,BIN 01101110,BIN
01110110,BIN 01111111,BIN 01111111,BIN 0
1110110,BIN 01101100,BIN 00101110
8210 FOR f=0 TO 7: READ a: POKE USR "k"+
f,a: NEXT f
8220 DATA BIN 00000010,BIN 00111010,BIN
10010010,BIN 11111110,BIN 11111110,BIN 1
0010010,BIN 00111010,BIN 00000010
8230 FOR f=0 TO 7: READ a: POKE USR "l"+
f,a: NEXT f
8240 DATA BIN 0,BIN 00111011,BIN 0001001
0,BIN 00111111,BIN 00111111,BIN 00010011
,BIN 00111011,BIN 0
8250 FOR f=0 TO 7: READ a: POKE USR "m"+
f,a: NEXT f
8260 DATA BIN 0,BIN 11011100,BIN 1100100
0,BIN 11111100,BIN 11111100,BIN 11001000
,BIN 11011100,BIN 0
8500 RETURN
```

Lines 8000-8500 The 'user-definable graphics' routine. Being a Your Spectrum aficionado, you'll probably realise that this routine is not as good as it could be. Take a look at previous issues and see if you can improve upon it.

```
9000 REM START
9010 LET lev=1: LET x$="[illegible]"
9020 BORDER 1: PAPER 6: CLS
9030 PRINT AT 2,0; INK 2;x$: BEEP .1,10:
 FOR f=60 TO 30 STEP -1: BEEP .05,f: NEX
T f: PRINT AT 5,0; FLASH 1;"[illegible]"
9040 PRINT AT 7,11;"SCOREING";AT 9,5;"10
 pts per hop forward";AT 11,4;"100 pts p
er frog put home";AT 13,3;"1000 pts for
all frogs home"
9050 PRINT AT 17,4;"STEER BY THE CURSOR
KEYS": GO SUB 8000: PRINT AT 20,3; FLASH
 1; BRIGHT 1;"  PRESS ANY KEY TO START
"
9060 IF INKEY$="" THEN  BEEP .1,RND*10+4
0: GO TO 9060
9070 BEEP .1,2: BEEP .1,4: BEEP .1,6: BE
EP .1,6: BEEP .1,4: BEEP .1,0
9080 CLS : RETURN
```

Lines 9000-9080 The 'initial start' routine. The level variable (lev) is set to equal one. Line 9030 prints the title on the screen in large letters and plays a tune before printing a flashing line across the screen. Lines 9040 and 9050 print the scoring and control key details. Line 9060 plays a random note while waiting for a key to be pressed.

```
9090 CLEAR : SAVE "FROGGER" LINE 1
```

Line 9090 The 'cassette saving' routine.

ADVERTISERS INDEX

YOUR SPECTRUM/STACK LIGHT RIFLE COMPETITION

THE MAN FROM S.P.E.C.C.Y.

LOAD

Forget about programming for this competition, we're just looking to pick your brains. The idea is to dream up a scenario for a game that will make good use of the Stack Light Rifle. Easy? Maybe, but Stack itself has covered most of the obvious games. In fact, included with the Light Rifle package are *High Noon, Shooting Gallery* and *Grouse Shoot* (and the games freaks amongst you will need no reminding as to what they involve!).

The Stack Light Rifle is a bit like the ultimate in joysticks — all you have to do is select a target on the screen, aim at it and pull the trigger. It's as simple as that! Your scenario, therefore, should involve something being shot at, whether it be a target in the traditional sense or some hideous meanie from outer space —and this is where we're looking for some imagination.

Ideas on cassette are OK, but really it's the *basis* of the game that's going to count — rather than the way you do it. Remember, what you've got to do is figure out the best way to incorporate a Light Rifle within the play.

Entries for the competition ought to include a short scenario of at least 200 words, and if necessary some sketches of the kind of screens you'd like to see. And, just a word of advice — the more fun you have devising the action, the more fun people are likely to have playing it. Think about it!

AIM

All ideas submitted will be sent to Stack Computer Services Ltd where they will be evaluated by a high power panel consisting of the Chairman, Managing Director and General Manager of the company. What they'll be looking for in your scenarios is originality, how you've used the Light Rifle within the program plan and the overall viability of the idea. Obviously, if colour and sound figure highly in the plot, then give the evaluators some idea of where these will be used.

Senders of the top 25 scenarios will each receive a Stack Light Rifle package. And it's even said that if your ideas are good enough, there's a chance that Stack might base a package around them. Who knows of the untold riches, etc, etc . . .



IAN McKINNELL

It's time to unplug your Spectrums and unlock your imaginations 'cause it's competition time again. James Bond, eat your heart out — if you'd handled a Stack Light Rifle on the little screen instead of a Walther Automatic on the big screen, you too would have been a sure-fire winner in the YS shoot-out!

FIRE!

Once you've sorted out what your ideal Stack software package would contain and have written it all down, simply bung it in an envelope and fire it off to the following address:

Stack Competition,
Your Spectrum,
14 Rathbone Place,
London W1

Latest date for entries to arrive is 25th May, 1984. The judgement of the panel will be final — no arguments, d'ya hear now!! YS

ANDROGENESIS

***Amongst the fear and loathing of the bright lights of London's West End,* Phil Z Manchester *gets hot on the trail of Prism's robot revolution optimist,* Grahame Daubney.**

The Hippodrome, Leicester Square: Strains of Richard Strauss and the *Star Wars* theme, scantily clad irrelevant go-go dancers, William Woolard (the tall one who used to be on *Tommorrow's World* now reincarnated as a computer salesman). The selling of microcomputers gets more and more like the recording business every day — the only thing missing is the stars. Programmers are usually too spotty to be sex symbols and the double-breast suited salesmen are just too sexless. The Prism group *know* this, so they import a few dancing girls and make their play in a trendy disco in the heart of London's West End night-life fantasy world.

A public relations person explains that it's the only place where they have enough room to show off the robots — so that's the reason why media hacks are to be found hanging around, drinking Prism's free booze under the laser lights! These proved to be a liablility later on. The digital stars, the Androbots that Prism is importing from the good ol' US of 'merica, sit at the end of an invisible infra-red thread remote from their Apple brains. Lasers and infra-red communications do not mix. Result = confused robots. Or maybe it was the go-go dancers; after all, "Robots are people too" says the slogan.

Instead of a demonstration of robotic dexterity, like humans, the Androbots sulk, their 'thoughts' restricted to pre-programmed recitations of 'Forward', 'Left', 'Right', 'Stop' when a convenient button on their plastic heads is pushed. All in all — not very impressive for £1500.

The scene shifts to Prism headquarters near Old Street station in north London. Topo, the sulking robot is still wandering up and down saying 'Left', 'Right', but doing little else. Grahame Daubney, Prism's director of new developments deftly sidesteps thrusting questions about Androbots, Micronet and Prism's future.

"I want to get to the situation where robots are usable by any member of the family," he explains. "He's still first generation but he is pretty sophisticated," Daubney added, pointing at the errant Topo.

Why 'he'?

"I call him 'he' because he has taken on a character. Atari has spent a lot on research trying to find out why *Pacman* was so popular. They discovered that it was because women liked it. The same goes for robots. Whenever we take Topo out somewhere the biggest reaction we get is from young women — they think he's cute.

"I think robots will bring more women into computers," Daubney goes on. Prior to joining new wave computer group, Prism, Daubney was at Atari. A remnant from those days still sits on his desk top — an ancient Atari computer. "I still use it for spreadsheet work — I'm used to it."

Daubney is convinced that robots are going to be the next big thing although the mass market is still some way in the future. The first assault is going to be on the education and development market. "I see it as being a way of bringing computers to more people, especially young kids. They get very bored with just screens."

"What we are trying to do is to find potentially useful applications rather than just playing around. We're not trying to advance technology — we want to advance the applications."

Down to nuts and bolts — when do we see robots for the Spectrum? Prism is the leading distributor for the 'handwarmer' so has a vested interest in selling add-ons. At the present stage robots are very much the computer peripheral (although Topo has a 'brother' with an on-board processor — Bob, which stands for 'brains on board').

"We're working on the Spectrum interface at the moment where it will act as a base station, through Interface 2. But we need to develop the software and a communications protocol. We could do the low level software in a few days but we want anyone to use it.

"We want to latch onto Logo as the model for controlling it." Logo is the language developed in the US by the Massachussetts Institute of Technology for teaching children about programming. One of its features is the 'turtle' — a sort of on-screen robot that can be commanded through the keyboard. Daubney also suggested that Forth may be a good language for handling robots but was noncommittal about producing a Spectrum interface.

He was also cagey about putting a launch date on any products that might bring robots into the Spectrum world, and was equally non-committal about future developments of the robot itself. Topo has no arms — it's just a sort of mobile speech synthesiser at present.

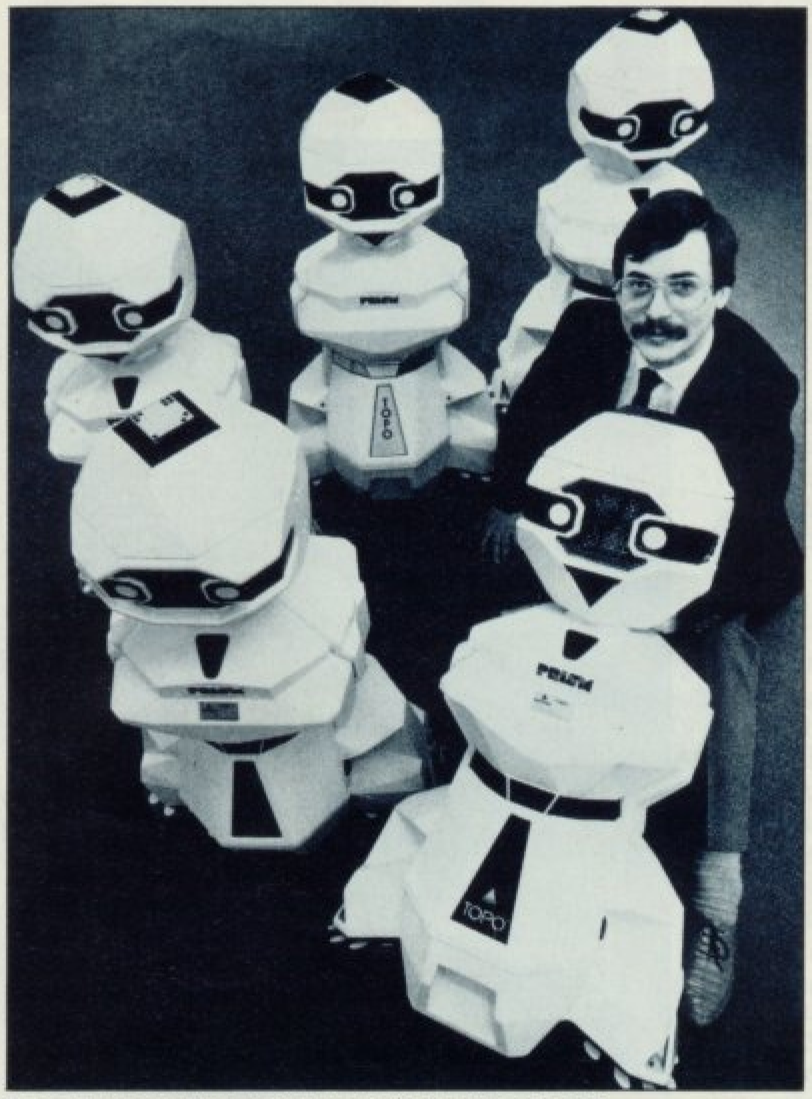
What about an 'arm' so that it can pick things up?

"That's one of the things we are approaching," he parried. "If someone comes to us with a good idea we may be able to help them and this is reflected in our recruitment policy. We're looking for people who maybe got their first computer a few years ago and have made their own add-ons. Some US software houses are recruiting artists and poets now to stimulate ideas."

All in all, the Androbots (incidentally, designed by the man who started Atari in a garage and sold it to Warner Brothers for a fortune before the rot set in) do not overwhelm one. They don't *do* much — especially in the company of laser beams and go-go girls.

The real clue to the reason for Prism's interest may, in fact, lie in the publicity handout for the beasts. One of the small number of uses that they mention, after mowing the lawn (unlikely) and cleaning the carpet (unbelievable), is as a 'promotional' aid. But it's a sad fact that Prism would probably have attracted as much attention with just the go-go girls (who didn't appear to be unduly affected by laser beams). A shame their link with new technology is so tenuous.

Oh, and don't hold your breath waiting for a robot to plug into the Spectrum. YS



Does Graham Daubney dream of electric sheep?

TONY SLEEP

GRAFPAD

for Sinclair Spectrum . . . Drawing into the future!

Now you too can have the benefit of drawing your own applications by the simple use of our NEW ZX Spectrum version of Grafpad!

THE GRAFPAD comes complete with Cassette Programme, Special Keyboard, Overlay, Plug, Light Pen and, of course, a comprehensive manual full of tips and instructions . . . all at the usual high quality expected from **BRITISH MICRO!**

STARTING UP is simple. Connect the Grafpad, switch on your Spectrum and connect the cassette player, type "LOAD" and press ENTER, then start the cassette player, place the keyboard overlay, and by the time you are ready, the "MENU" will flash on to your screen! Usage? Well there is . . .

AS MANY USES AS YOU CAN IMAGINE by first following our easy manual, then, in no time, you will be able to utilise as you Think best!

A UNIQUE PRODUCT that will improve your skills and give you endless hours of excitement!







Size
25mm height
55mm width
260 depth
weight 1.2k (gross)



- ★ Dealer inquiries welcomed
- ★ Special discounts to educational authorities & government departments
- ★ Export inquiries invited

HOW TO ORDER
BY TELEPHONE: (0923) 48222
If you are an American Express, Barclaycard, Diners Club or Access Card Holder simply telephone us giving your Card No., Name, Address and item(s) required and your order will be despatched within 48 hours!

BY POST:
Simply fill in the coupon enclosing your cheque/P.O. made payable to: BRITISH MICRO or use the special section for Credit Card Holders and post to the address below.
OVERSEAS ORDERS:
Please add £10 per item ordered.

Post to: BRITISH MICRO, UNIT Q2, PENFOLD WORKS, IMPERIAL WAY, WATFORD, HERTS. WD2 4YY.

Please send me GRAFPAD for: SINCLAIR SPECTRUM
☐ BBC MODEL B ☐ COMMODORE 64
If you require details of the above versions, please tick.

Qty.	Item	Ex. VAT	Inc. VAT	Total
	Grafpad Complete	£125.00	£143.75	
			Postage, Packing & Insurance	£5.00
			TOTAL	£

I enclose my cheque for /P.O. £...........

I prefer to pay with my American Express, Barclaycard, Diners Club, Access Card.
(Please delete whichever is not applicable)

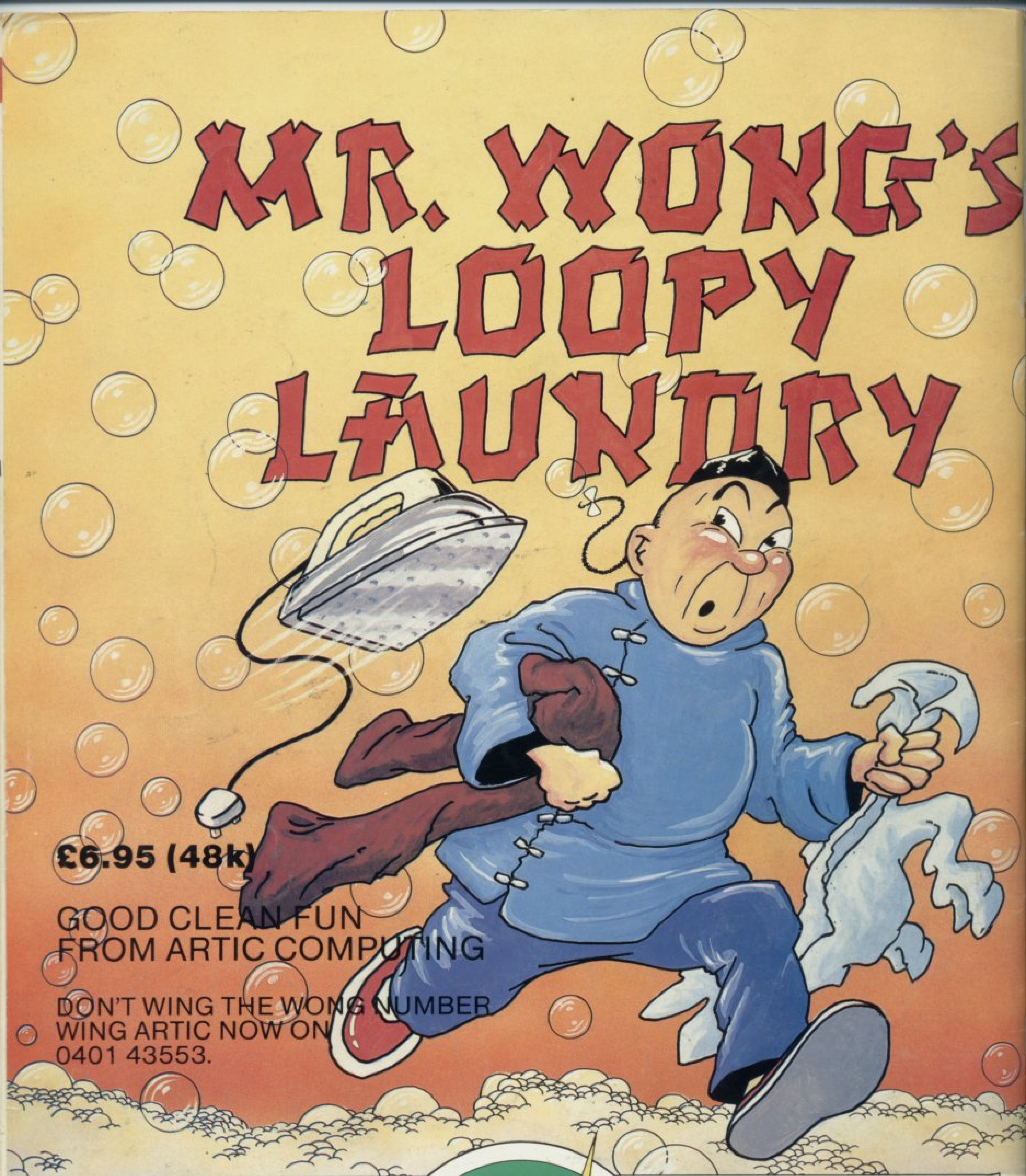
Card No. ......
Signature ......
Name ......
Address ......

Address above must be the same as Card Holder

BRITISH MICRO
A HEGOTRON GROUP COMPANY

Unit Q2, Penfold Works,
Imperial Way, Watford, HERTS WD2 4YY
TEL: (0923) 48222. TELEX: 946024

★ INQUIRE ABOUT BBC VERSION.



ARTIC COMPUTING LTD

Main Street, Brandesburton, Driffield YO25 8RL

Cheques and PO's payable to Artic Computing Ltd.
Dealer enquiries welcome.
Access and Barclaycard welcome.

WARNING: These programs are sold according to ARTIC COMPUTING LTD's terms of trade and conditions of sale. Copies of which are available on request.

Send S.A.E. (9in. × 6in.) for FREE catalogue of our wide range of programs.

artic Computing Ltd

To: Artic Computing Ltd, Main Street
Brandesburton, Driffield YO25 8RL

Please supply

*Cheque for total amount enclosed £

Name

Address

*Access/Barclaycard No.

*Please delete or complete as applicable